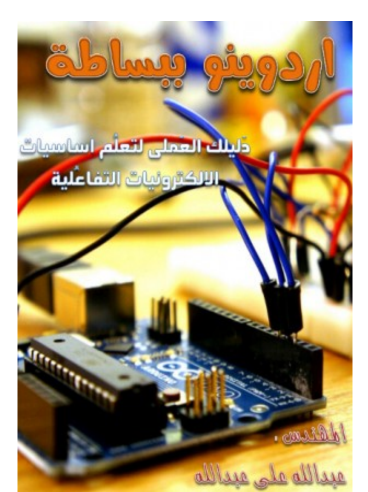
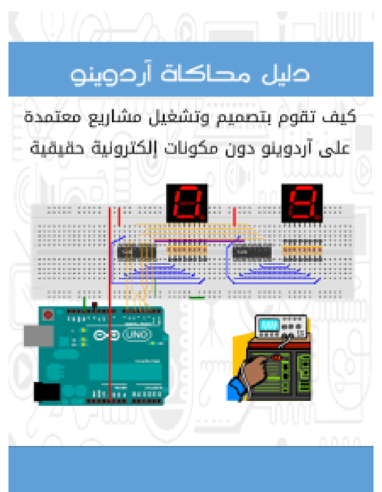

اردوينو ببساطة
دَليلك العَملى لتعلُم اساسيات
الالكترونيات التفاعُلية
المهندس :
عبدالله على عبدالله

# سلسلة " تعلم ببساطة "

تهدف سلسلة كتب" تعلم ببساطة " إلى توفير علوم الإلكترونيات الحديثة باللغة العربية بصورة مجانية ومفتوحة المصدر مع الحفاظ على المبدأ الأساسي "**البساطة**" في شرح المعلومات، جميع الكتب موجهة للعامة من هواة الإلكترونيات و طلبة الكليات الهندسة.






# رخصة الكتاب



يمكنك تحميل السلسلة من الموقع http://simplyarduino.com/?page_id=889

للتواصل مع المؤلف

abdallah.ali.abdallah.elmasry@gmail.com

حساب Facebook

www.facebook.com/abdallah.ali.elmasry

حساب linkedIN

eg.linkedin.com/in/abdallahali

عبدالله علي عبدالله

1434 هـ الموافق 2012 م.

# اهداء

الى والدى العزيز....

الى والدتى العزيزة....

الى كل من ساهم فى نشر المعرفه بحريه

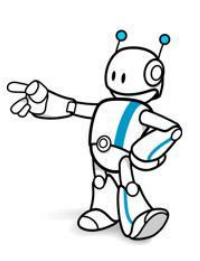
| | |
|---|---|
| أيمن النعيمي | www.networkset.net |
| أحمد محمد أبوزيد | www.aabouzaid.com |
| رءوف شبايك | www.shabayek.com |
| مجتمع لينكس العربي | linuxac.org |

# فهرس الكتاب

قال الله ﷻ في القران الكريم:

بِسْمِ ٱللَّهِ ٱلرَّحْمَٰنِ ٱلرَّحِيمِ

ٱقْرَأْ بِٱسْمِ رَبِّكَ ٱلَّذِى خَلَقَ ﴿١﴾ خَلَقَ ٱلْإِنسَٰنَ مِنْ عَلَقٍ ﴿٢﴾
ٱقْرَأْ وَرَبُّكَ ٱلْأَكْرَمُ ﴿٣﴾ ٱلَّذِى عَلَّمَ بِٱلْقَلَمِ ﴿٤﴾ عَلَّمَ ٱلْإِنسَٰنَ
مَا لَمْ يَعْلَمْ ﴿٥﴾



## الفئات المستهدفه من الكتاب

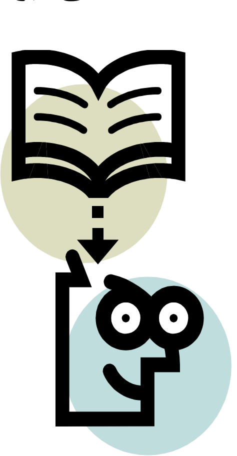
الى كل من يريد تعلم تقنيات التحكم الآلى بسهوله و يسر سواء كان لك معرفه تقنيه او خبره بالمتحكمات الدقيقه او حتى ما زلت فى بدايه تعلُم هذا المجال و الى كل من لديه الشغف لمعرفه الجيل الجديد من المتحكمات الدقيقه مفتوحه المصدر

## هل احتاج الى خبرة فى مجال الإلكترونيات و البرمجة حتى اقرأ هذا الكتاب ؟



- بعض المعرفه باساسيات الالكترونيات و البرمجه محبذه جدا لاكنها ليست شرط ضرورى للتعلم فلقد قمت بكتابه المعلومات بأسلوب سهل و ميسر قدر الامكان و يمكنك اذا واجهت صعوبه ما ان تراسلنى على البريد الالكترونى ☺

## ما الذى سوف اتعلمه فى هذا الكتاب؟؟



- ✓ كيف تصمم الإلكترونيات التفاعلية بنفسك
- ✓ العمل على بيئة تطوير اردوينو Arduino IDE
- ✓ التحكم الالكترونى لكل ما هو حولك من اجهزه الكترونيه




**سيجعلك الكتاب** تكتسب المهارات الأوليه لفهم الأكواد البرمجيه المستخدمه فى العديد من المشاريع التى تعتمد على **اردوينو** ، ولقد اضفت العديد من المصادر الخارجيه التى تحتوى على العديد من المشاريع الرائعه و التى يمكنك ان تنفذها بنفسك بعد قراءة الكتاب

## كيف تم تصميم الكتاب ؟



**تم تصميم** الكتاب بأسلوب يعتمد على الشرح بالأمثله Examples و حيث يتم تقديم المعلومات على هيئه أمثله صغيره و يقدم كل مثال جزء جديد فى تعلم لغه برمجه اردوينو ، كما ستجد في نهاية كل فصل صفحه مخصصه لكى تكتب ملاحظاتك الشخصية.

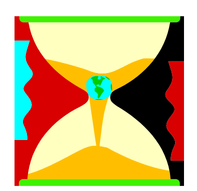
اذا كنت من مستخدمي ويندوز يمكنك أن تكتب داخل ملفات الكتب الإلكترونية من نوع PDF باستخدام برنامج Foxit PDF و اذا كنت تقرأ الكتاب باستخدام لوح الكترونى Tablet مثل Ipad يمكنك استخدام برنامج Good Reader مع ملاحظه ان الكتاب متوافق مع جميع اجهزه التابلت المختلفه التى تمتلك شاشه بقياس 7 بوصه او اكبر .

# الفصل الأول

## مقدمه الى عالم اردوينو و المتحكمات الدقيقه

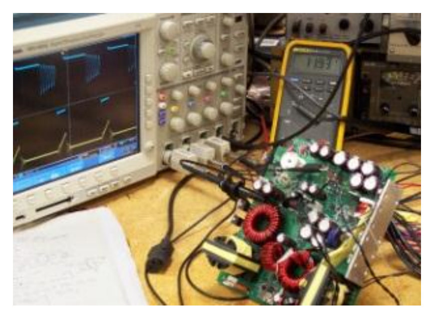
## Introduction to Arduino & microcontrollers



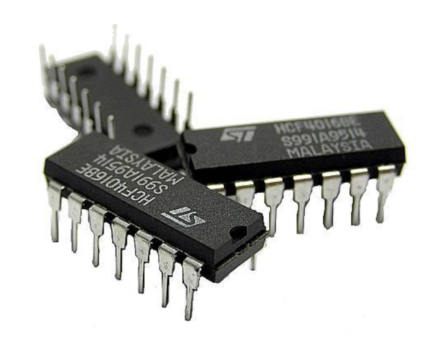


**منذ زمن** ليس ببعيد كان العمل على صناعه دائره الكترونيه للقيام بوظيفه معينه يعنى بناء تصميم الكترونى معقد من مكونات مثل المقاومات، المكثفات، الملفات، الترانزستور...الخ.

**كانت** الدوائر الالكترونيه ثابته التصميم و اعاده تغير او تعديل جزء بسيط فيها كان يعنى الكثير من عمليات معقده مثل اللحام و قطع الاسلاك و اعاده النظر فى المخططات الالكترونيه و الكثير من الامور المزعجه و التى ادت إلى اقتصار وظيفه تطوير المنتجات الالكترونيه على مجموعه من المهندسين المتخصصين فقط.



**بفضل** التطور التكنولوجى فى مجال اشباه الموصلات و اختراع الدوائر المدمجه Integrated Circuits (IC) اصبح من الممكن وضع دائره الكترونيه كامله على شريحه صغيره حجمها قد لا يتجاوز رأس الدبوس حتى انه فى الوقت الحالى هناك دوائر الكترونيه حجمها يقدر بالنانو متر Nano Meter و التى لا يمكن روئيتها الا بأستخدم مكبرات ضوئيه خاصه.



**كما ادى** تطور Integrated Circuits الى ظهور جيل خاص من الدوائر الالكترونيه يسمى المتحكمات الدقيقه **Micro Controllers** و هى اشبه بكمبيوتر مصغر قابل للبرمجه لاداء مجموعه من الوظائف مثل قرائه درجه حراره، التحكم فى محرك كهربى، او حتى اداره خطوط الانتاج فى المصانع الكبرى، و كل ذلك يتم ببساطه عن طريق اوامر برمجيه و بذلك تحولت تقنيه صناعه الدوائر و الانظمه الالكترونيه من التصميم الالكترونى البحت المعتمد على المكونات الصلبه فقط الى اوامر برمجيه يمكن لاى فرد ان يكتبها و يصممها بنفسه بسهوله و يسر.



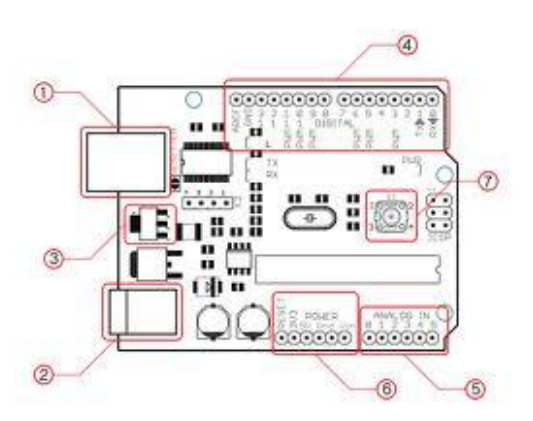



**تتميز** المتحكمات الدقيقه ايضا بأمكانيه التغير و التعديل فى اى وقت ، بكل بساطه لو اردت ان تغير شيئا ما فى مشروعك يمكنك ذلك بالتعديل فى السطور البرمجيه و اعاده وضع الاوامر الجديده على المتحكمه الدقيقه و تجربتها اكثر من مره و هكذا الى ان تصل بمشروعك للهدف المنشود ...



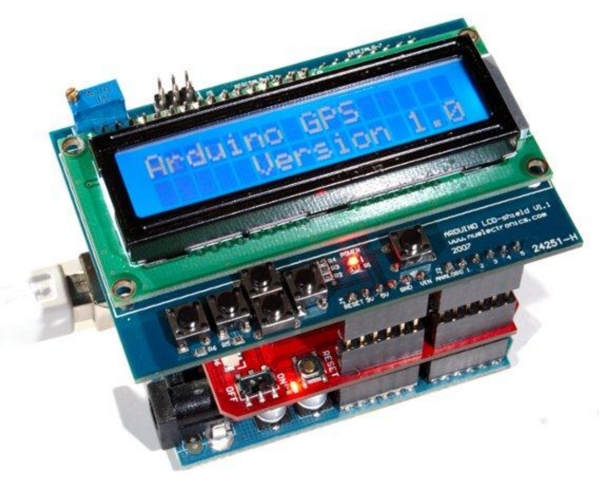
**اردويـنـو Arduino** هـى بـورده الكترونيـه مفتوحـه المصـدر Open Hardware لتطـوير الكثير من الافكـار و المشـاريع المتعلقـه بـالتحكم الآلى بصـوره سـهله و بسـيطه عـن طريـق استخدام لغه برمجه مفتوحـه المصـدر Arduino C و يـتم برمجـه المتحكمـه الموجـوده علـى البـورده بأسـتخدام برنـامج خـاص يسـمى Arduino IDE: Integrated Development Environment و الــذى يتــوفر مجانــا للتحميــل مــن موقــع اردوينــو الرســمى http://arduino.cc

## ما معنى ان اردوينو مفتوح المصدر ؟؟



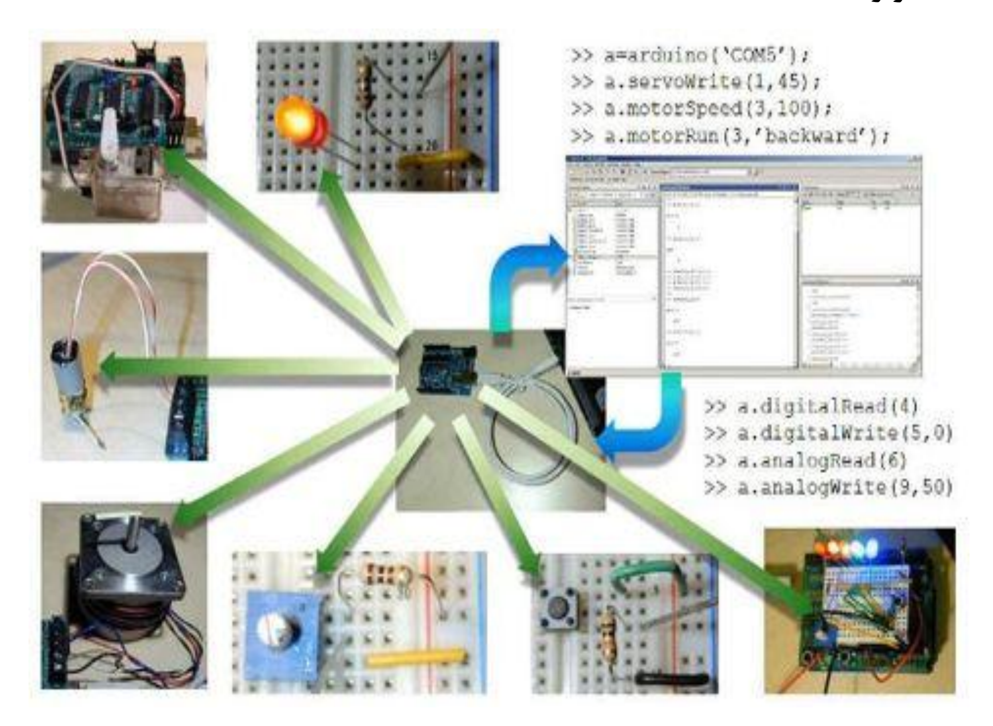

**يعنى انـه** يمكنـك الاطلاع و التعديل علـى التصـميمات الهندسـية و الشـفرات المصـدرية Source Codes لكل من بوردات **اردوينو** المختلفة Arduino Boards و Arduino IDE بما يتناسـب معـك و يمكنـك ايضاً تطوير لغة برمجه Arduino C بحريه تامه و الاطلاع على الشفرات المصدرية الخاصة بها كما ان كل هذه المميزات و البرمجيات مجانيه تماما علـى غرار بعض البيئات التطـويره مثل **Mikro C** و التى تتطلب منك شراء رخصه مكلفه تصل فى بعض الاحيان الى الآف الدولارات لاستخدامها.

**ما يميز بوردات اردوينو** عن باقي البوردات التطويريه للمتحكمات الدقيقه الاخرى Micro Controllers Development boards هو مدى السهوله التعامل معها و بساطه اللغه البرمجيه و التى عمل فريق من ايطاليا على تطويرها منذ عام 2005 حتى الان ، و لقد تم اشتقاق لغه اردوينو البرمجيه من لغه processing ولغه "سى C" و التى تعد اساس لغات البرمجه الحديثه و صاحبه ثوره تقنيه البرمجيات.




**قـد يظـن** الـبعض ان اردوينـو مصـمم للـهواه فقـط لاكـن هـذا لـيس صـحيح لقـد تم تطـويره ليناسـب جميـع المسـتويات ابتـداء مـن الهـواه و انتهـاء بالمشـاريع المتطـوره و الـدليل انه هنـاك مميزات جباره تجعل اردوينو على قمه المتحكمات الدقيقه و هى امكانيه دمجه فى مشاريع يتم برمجتها بلغات هندسيه متطوره مثل MATLAB و لغه Java حيث ستجد مكتبات برمجيه فى جاهزه للغه الـMATLAB و الجافا خاصه بالتعامل مع اردوينو، و ستجد فى صفحه المراجع بعض الموارد التى تعلمك كيفيه استخدام الاردوينو مع المات لاب، كما يمكنك ايضا استخدام لغه Java و لغه VB.NET فى التواصل مع بوردات اردوينو المختلفه

**على سبيل المثال**: يمكنك عمل دائره استشعار للطقس و الحراره بالاردوينو و ارسا ل البيانات الى MATLAB على الحاسب الآلي لاجراء التحليلات الحسابيه المتطوره المتوفر بلغه MATLAB.




**من اكثر** المشاريع التى اثارت اعجابى كانت مستكشف زلازل بأستخدام الاردوينو و لغه برمجه MATLAB و كانت من تصميم شاب فى السادسه عشر من عمره فى الولايات المتحده حيث قام بعمل وحده استعشار و تحليل للزلازل بتكلفه لا تتجاوز 80 دولا ر باستخدام بعض المستشعرات و بورده **اردوينو** فقط مع العلم ان الدوائر الالكترونيه المماثله قد تتكلف اكثر من 2000 دولار س

# انواع بوردات اردوينو Arduino Boards

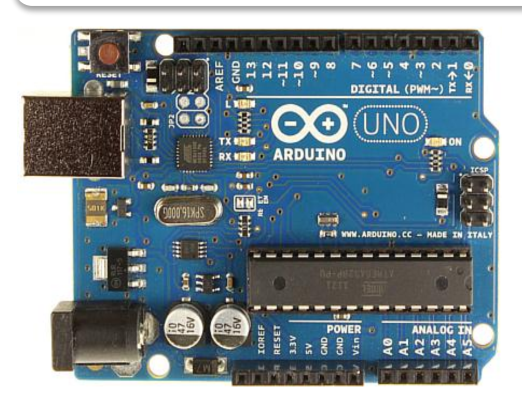






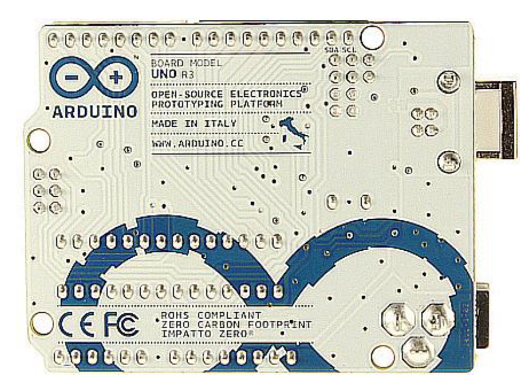

Arduino UNO, Arduino Mega, Arduino Nano, Arduino Mini, Arduino Lilypad, Arduino Demulive, Boarduino

**تختلف** البوردات عن بعضها البعض من ناحيه عدد المخارج و المداخل و التى تحدد عدد الاجهزه التى يمكن التحكم بها و عدد الحساسات Sensors التى يمكن دمجها مع البورده و كذلك نوع المتحكمه الدقيقه و سرعه المعالج الموجود بداخلها و امكانيه تبديلها ام لا و سوف نتناول فى هذا الكتاب بورده **Arduino Uno**



# نظره عامه على دائره Arduino Uno

**دائرة الكترونيه** صغيره تستخدم في برمجه متحكمه من شركه اتمل ATmega328 و توفر هذه الدائرة منافذ لتوصيل المكونات الالكترونيه الى المتحكمه مباشره عن طريق 14 (مدخل | مخرج) من النوع الرقمي Digital In/out من هذه ال14 يوجد 6 يمكن استخدامها كمخارج PWM أو ما يعرف **بالتعديل الرقمي المعتمد على عرض النبضة**(Pulse-Width modulation) و سوف نتحدث عن هذه الخاصيه في فصل كامل يسمى الدخل و الخرج التماثلي Analog Inputs & Outputs



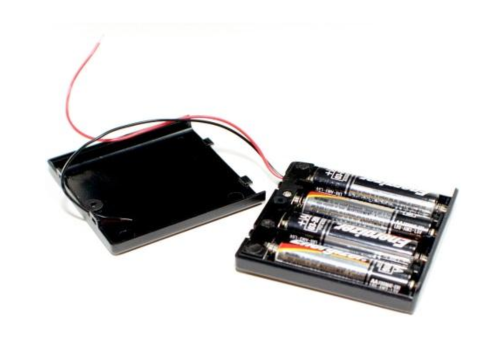
**أيضا** تحوي الدائرة على مهتز كريستال Crystal Oscillatorبتردد 16MHz، بالاضافه الى مدخل USB من أجل التواصل مع الحاسب، و هناك مدخل للطاقة منفصل ، بالاضافه الى ICSP header والذي يعتبر طريقه اضافيه لبرمجة المتحكمة وهي لا تزال موصلة بالبورده (بخلاف الUSB) و يمكنك ان تعتبر بورده اردوينو هذه بورده تطوير و برمجه مصغره و مُهيئه للاستخدام المباشر Development Boardفهي تقريبا تحوي كل ما تحتاج لكي تعمل عليها سواء عن طريق منفذ الـUSB أو عن طريق مصدر خارجي للطاقه مثل البطاريه.

## امداد الدائره بالطاقه Power up




**يمكنك** امداد الدائرة بالطاقة إما من خلال منفذ الـUSB فقط أو عن طريق استخدام مصدر خارجي للكهرباء كمحول AC\DC ليمد الدائرة بالجهد اللازم للعمل او حتى عن طريق بطارية 9 فولت او 4 بطاريات 1.5 فولت حيث يتم توصيل طرفي البطارية الى مدخل الارضي Gnd و الـ Vin في الدائرة.



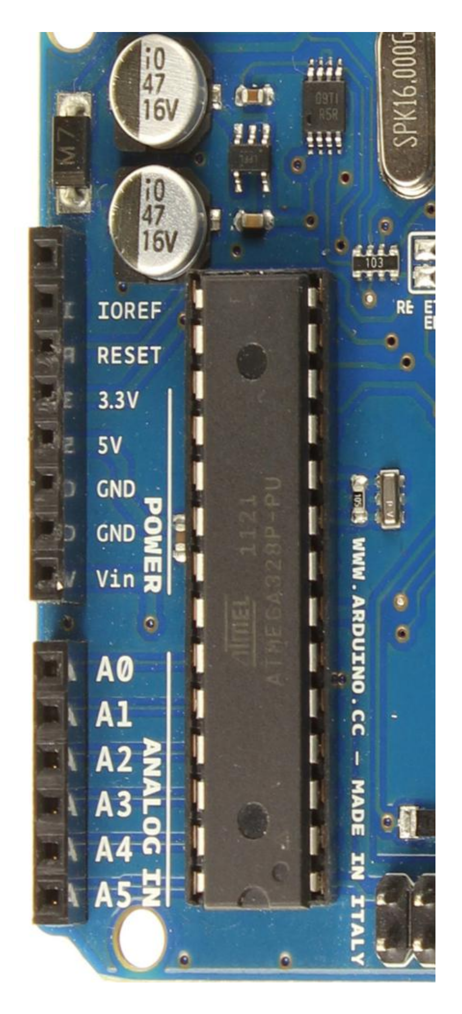






**تستطيع** الدائرة العمل على جهد يتراوح بين ال 6 –20 فولت، لكن يجب الانتباه حيث انه اذا قمنا بتأمين جهد أقل من 7 فولت فإن المخرج المتحكمة Pin 5V قد لا يستطيع تأمين جهد خرج يبلغ الـ5 فولت المطلوب وقد يؤدي الى عدم استقرار الدائرة، اما اذا قمنا بتزويد الدائرة بجهد أعلى من 12 فولت فإنه قد يؤثر على عنصر تنظيم الجهد regulator voltage ويؤدى الى ارتفاع درجة حرارته مما يؤدي الى تلف البورده ، لذا فإن مجال الجهد الذي يفضل استخدامه هو من 7 الى 12 فولت.

# مخارج و مداخل الطاقه الكهربيه للمتحكمة

## Power Inputs/Outputs




- **Vin** : جهد الدخل عندما نستخدم مصدر طاقة خارجي، يمكننا تأمين الجهد من خلال هذا المدخل ، إذا كنا نقوم بتأمين الطاقة للدائرة من خلال مدخل المحول يمكننا الوصول له من خلال هذا المدخل أيضاً.

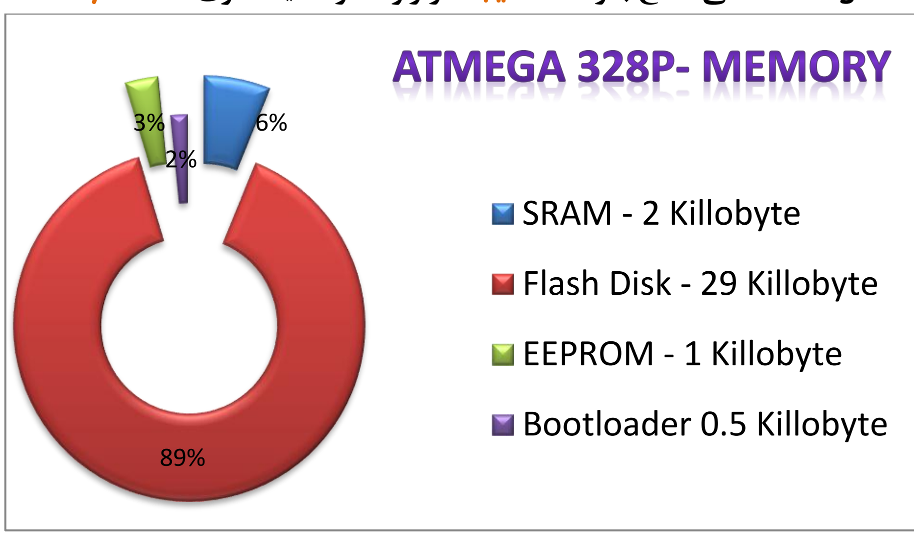
- **V5** : جهد منتظم يستخدم لتأمين الطاقة للعناصر المستخدمه على الدائرة و سوف نستخدمه لتوفير الطاقه للقطع الالكترونيه التي سنضيفها، قد يأتي هذا الجهد من خلال Vin عبر منظم جهد داخلي أو تأمينه من خلال منفذ الUSB أو أي مصدر جهد منتظم بقيمه 5 فولت.

- **V3.3** : مصدر للجهد بقيمة 3.3 فولت مؤمن من قبل منظم الجهد الداخلي للدائرة و أقصى قيمة لسحب التيار من خلال هذا الخط هو 50 ميلي أمبير.

- **GND** : الخط الارضي.

# المعالج الدقيق و الذاكره Microprocessor

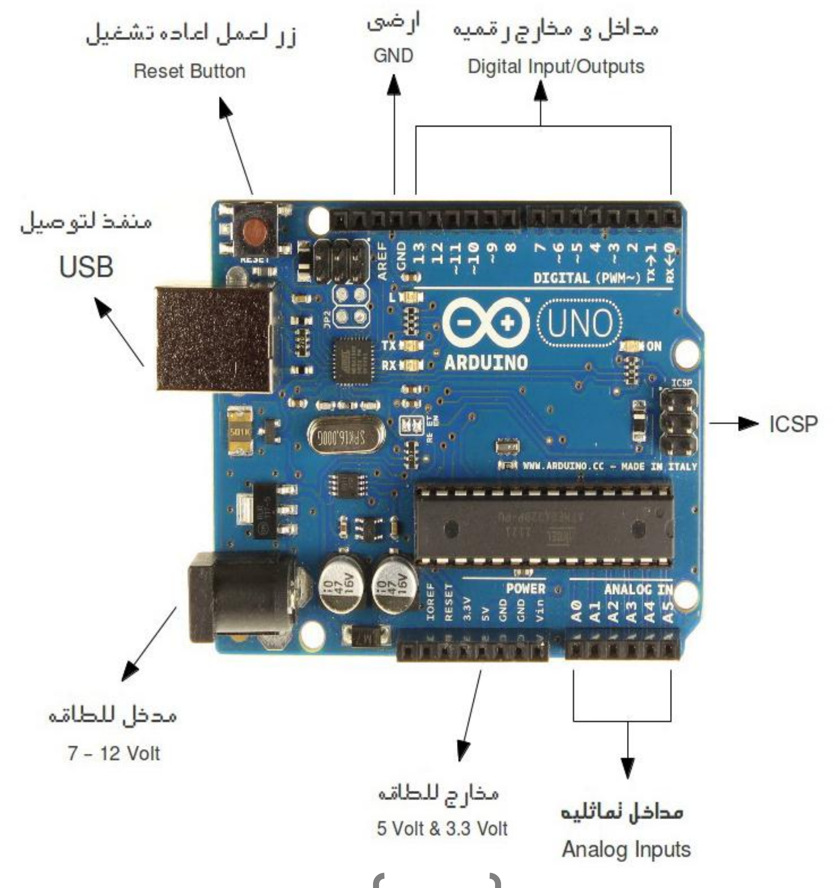
المتحكمات الدقيقه اشبهه بوحده حاسب آلى صغيره الحجم و تحوي المتحكمه الدقيقه ATmega328 على معالج بسرعه 16 ميجا هرتز و ذاكرة كليه تساوى 32 Kilo Byte




- Boot loader : السوفت وير المسؤل عن كيفيه فهم الدائره للغه Arduino C
- SRAM: تعتبر الذاكره المُستخدمه فى تسجيل المتغيرات بصوره مؤقته.
- Flash Disk : مساحه تخزينيه تستخدم فى تخزين البرنامج الذى سنكتبه لتشغيل المتحكمه ، فى الوهله الاولى قد يبدو هذا الرقم صغير جدا لكنه فى الحقيقه كافى لكتابه الكثير من الاوامر كما سنرى فى الامثله القادمه.
- EEPROM : الذاكره المسؤله عن تسجيل بعض المتغيرات بصوره دائمه داخل المتحكمه و تظل محتفظه بقيتمها حتى بعد فصل الكهرباء و يمكننا ان نعتبرها مثل السواقه Hard Disk فى الكمبيوتر الشخصى.

## مداخل و مخارج التحكم (I/O) Input & Output Pins

**يمكن** تخصيص الخطوط الرقمية الاربعه عشر (14 Digital Pins) كمداخل أو مخارج وذلك باستخدام الاوامر البرمجيه كما سنرى فى الفصل الثانى و تعمل هذه الخطوط على جهد اقصاه 5 فولت وكل خط يمكن أن يؤمن سحب للتيار بحدود ال 40 ميلي أمبير، و هناك 6 خطوط دخل تماثلية Analog ومعنونة من A0 الى A5،بشكل افتراضي تستطيع هذه المداخل قياس جهد من صفر حتى 5 فولت.



## ملاحظات شخصيه :

هذه الصفحه مخصصه لكتابه ملاحظاتك الشخصيه عن الفصل الاول :

اكتب ملاحظاتك
هنا

# الفصل الثاني: التجهيزات

## اولا: المكونات الماديه Hardware & Tools



## كيف تبدأ استخدام اردوينو و ماذا ستحتاج

سوف نستخدم بورده من نوع Arduino UNO و ستحتاج للمكونات التاليه كبدايه:

(ملحوظه: فى بعض الامثله ستحتاج اضافات لهذه المكونات)

### Arduino Uno V.3

هى البورده المسئوله عن استقبال الأوامر البرمجيه من الحاسب الآلي و كتابتها فى ذاكره المتحكمه الدقيقه ATmega328 الموضوعه بداخلها




### USB printer cable (1 Meter)



كابل طابعه لتوصيل البورده بالحاسب الآلي، يمكنك شراء اى نوع و يفضل ان يكون قصير قدر الامكان حتى لا يسبب ازعاج اثناء العمل ،طول " واحد متر " سيكون اكثر من كافي



### لوحه التجارب (Breadboard)

بورده خاصه تستخدم لتركيب و فك المكونات الالكترونيه عليها بسهوله دون الحاجه الي لحام ، يمكنك شراء اى حجم ويفضل Half Sized Breadboard



### دايود ضوئي (الوان مختلفة)Colored Leds



بعض الليدات و هى اشبه بالمصابيح الصغيره و تعمل على تحويل التيار الكهربى الى ضوء، يفضل ان تشترى 10 ليدات بألوان و احجام مختلفه و تتوفره بألوان مثل:الاحمر، الأخضر، البرتقالي، الأصفر، الابيض و الازرق

### مقاومات Resistors



سنحتاج الى مقاومات للتحكم فى التيار الكهربى على مداخل و مخارج المتحكمه الدقيقه و سنحتاج المقاومات التاليه:

Resistor 560 Ohm (x5)
Resistor 10 Killo-Ohm (x5

### AVO Meter (multi-meter device)



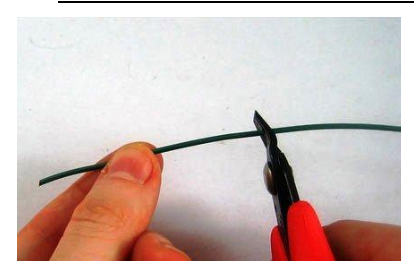

(اختياري) جهاز قياس كهربي متعدد الاستخدامات و يستطيع قياس المقاومات و فرق الجهد و شده التيار الكهربى

### بعض الاسلاك للتوصيلات بين المكونات Wires

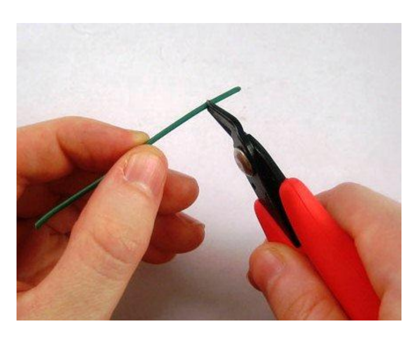


ستستخدم فى توصيل المكونات بعضها على لوحه التجارب و ستتعلم كيف نقوم بتجهيز السلك بنفسك كما سنرى فى الصفحه التاليه.

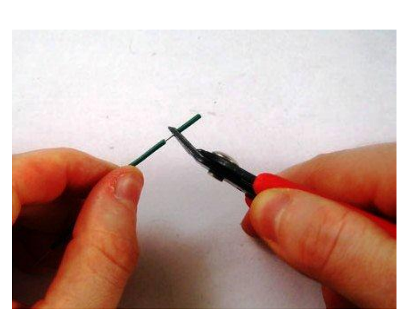
# كيف تُجهز أسلاك التوصيل



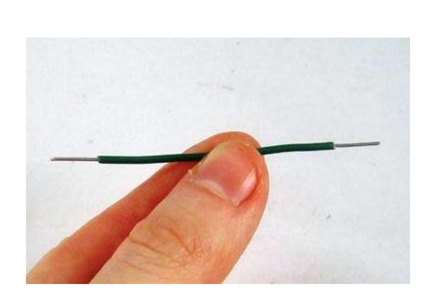
يمكنك الحصول على وصلات عاليه الجوده عن طريق استخدام قطعه من كابل Cat5 المستخدم في وصلات شبكات الحاسب الآلي و تستطيع شراءه من محلات مستلزمات الكمبيوتر و الشبكات .



قم باستخدام قشاره الاسلاك و قص واحد سنتي متر من الجهتين للسلك المراد تجهيزه كما في الصوره الاولي، ثم كرر نفس العمليه مره اخرى ولكن باطوال مختلفه حتى تحصل على مجموعه اسلاك جاهزه للاستخدام



اذا كنت لا تفضل ان تقوم بهذه العمليه يمكنك شراء مجموعه اسلاك مقشره جاهزه للاستخدام مع ال breadboard و ستجدها في محلات المكونات الالكترونيه







**اذا كنت** فى مصر و ترغب فى شراء تلك المكونات هناك 3 مواقع مصريه على الانترنت تستطيع طلب تلك المكونات منها و سيتم شحنها لباب منزلك فى غضون يومين و سيتم اضافه تكلفه الشحن على سعر المكونات.

http://www.fut-electronics.com
http://egyrobots.com
http://www.ram-e-shop.com

اما اذا كنت تفضل زياره بعض المحال بنفسك



يمكنك الذهاب الى محل رام الكترونيكس RAM Electronics الموجود فى " **باب اللوق** " و محل فيوتشر الكترونيكس Future Electronics الموجود امام **كليه هندسه عين شمس** فى ميدان **عبده باشا** فى العباسيه .



و اذا كنت فى احدى الدول العربيه او اى دوله اخرى لا يتوفر بها محلات تبيع هذه المكونات (خاصه **اردوينو**) يمكنك زياره الموقع التالى و ستجد عليه الكثير من المنتجات الخاصه **بادروينو** مع الشحن المجانى

http://www.buyincoins.com

كما يمكنك زياره قائمه الموزعين العالميين **لاردوينو** لتبحث عن موزع محلى قريب منك عبر الرابط التالى:

http://arduino.cc/hu/Main/Buy




## محتويات علبه Arduino Uno

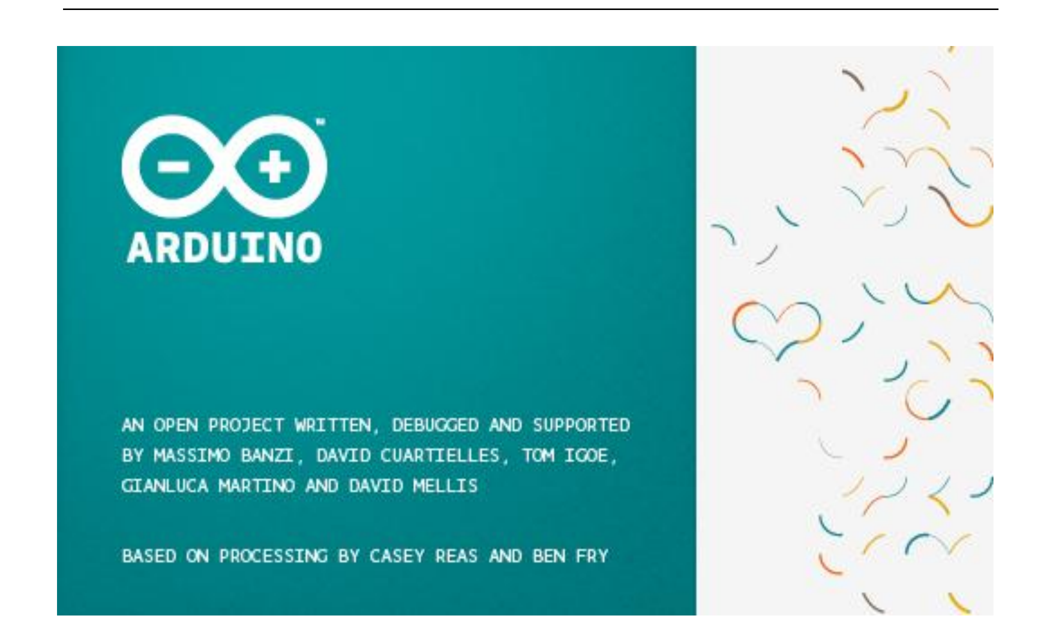
ملحوظه قد تختلف مكونات العلبه من بلد الى اخرى على حسب المورد و رقم اصداره البورده كما قد تحتوى على ملصقات اضافيه لاردوينو

# ثانيا: تجهيز البرمجيات



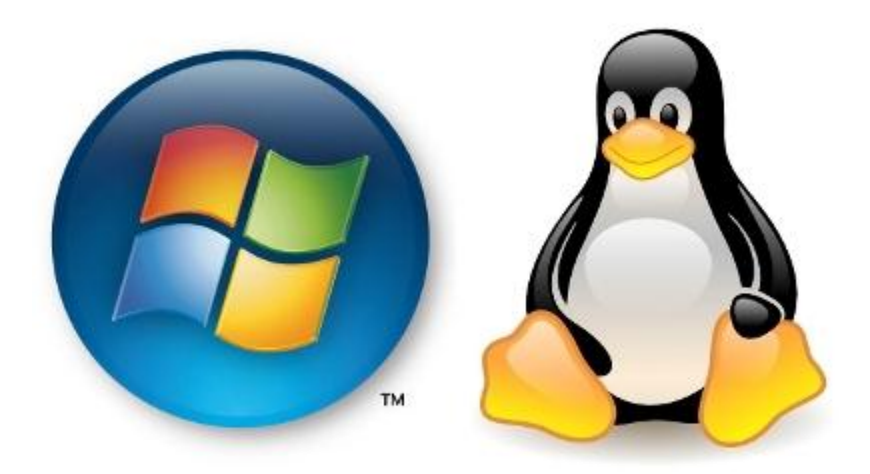
# Software's

# بيئه تطوير اردوينو Arduino (IDE)




**تعتبر بيئه التطوير Arduino IDE الاداه المستخدمه فى كتابه الاكواد البرمجيه بلغه Arduino C و تحويلها بعد ذلك الى صيغه تنفيذيه يمكن وضعها على المتحكمه الدقيقه الموجوده على البورده .**

**تتميز بيئه تطوير اردوينو بالبساطه و السهوله فى التعامل فهى تكاد تخلو من اى تعقيدات فى المظهر العام و تحتوى فقط على ما يحتاجه المبرمج ليبدأ تطوير برامج بلغه اردوينو سى Arduino C كما انها تستخدم فى نفس الوقت لرفع البرنامج مباشره الى المتحكمه الدقيقه و بذلك لن تحتاج الى برنامج اخر مخصص لرفع الصيغه التنفيذيه للبورده.**



**سـيتم الشـرح** علـى بيئه تطـوير اردوينـو النسخه الخاصه **بويـنـدوز** بأعتباره اكثر انظمه التشغيل شيوعا و شهره بين المستخدمين بالرغم من انـى انصح بشـده بأسـتخدام لينـكس لانه مجانى و سريع و آمن بالاضافه الى انه **مفتوح المصدر** Open Source ، و فى جميع الاحوال يمكنك العمل على اى نظام تشغيل تحب.




**من يريد** التعرف اكثر على نظام تشغيل لينكس هناك كتاب عربى مجانى **[ اوبنتو ببساطه ]** من تأليف:
**أحمـد محمـد أبوزيـد** و يشـرح اسـتخدام نظـام تشـغيل **لينكس ابونتو** Ubuntu باسلوب سهل و ممتع و يمكنك تحميله مجانا من على الموقع الرسمى للكتاب :
http://www.simplyubuntu.com/

# تنزيل بيئه التطوير Arduino IDE



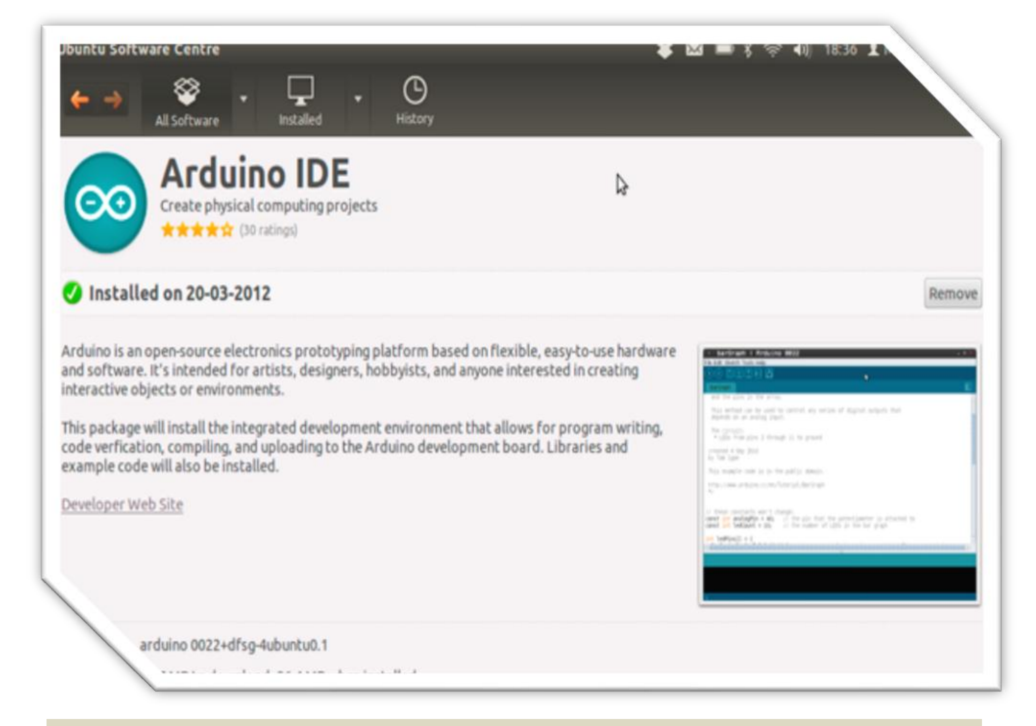
**تتـــوفر** بيئـــه التطـــوير البرمجيـــه لجميـــع انظمـــه التشـــغيل المختلفة Windows, Linux, MAC ويمكنك تحميلها من الموقع الرسمى لمطورى اردوينو على الرابط التالى:

http://arduino.cc/hu/Main/Software

**بالنسبه** لمستخدمى ويندوز او ماك قم بتنزيل الاصداره الخاصه بهما من الرابط اعلاه و ستجد ملف مضغوط، قم بفك الضغط بأستخدام احدى البرامج مثل 7zip ثم قم بتشغيل ملف arduino.exe

arduino-0009
File Edit View Favorites Tools Help
Back Search Folders
Address C:\Documents and Settings\ladyada.ENTHEOGEN.000\Desktop\arduino-0009 Go
bootloader
bootloader168
drivers
examples
java
lib
reference
sketchbook
tools
arduino.exe
cygiconv-2.dll
cygwin1.dll 1005.19.0.0 Cygwin® POSIX Emulation DLL
ICE_JNIRegistry.dll
jikes.exe
libusb0.dll 0.1.12.1 LibUSB-Win32 - DLL
readme.txt Text Document 9 KB
run.bat MS-DOS Batch File 1 KB
rxtxSerial.dll

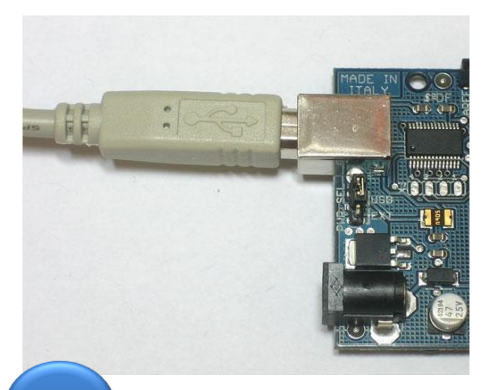
**لمســـتخدمى** نظـام ليــنكس اوبنتــو Ubuntu يمكنــك اســتخدام مركــز البرمجيــات Software center فقط ابحث عن Arduino IDE ثم اضغط زر Install و بالنسبه لمستخدمى انظمه لينكس الاخرى مثل فيدورا fedora و نسخه اوبـن سـوزى Open SUSE ستجدو بيئه التطوير موجوده فى مركز البرمجيات الخاصه بالتوزيعه ان لم تستطع ايجاده يمكنك تحميل الملف الخاص بلينكس من على الموقع الرسمى.



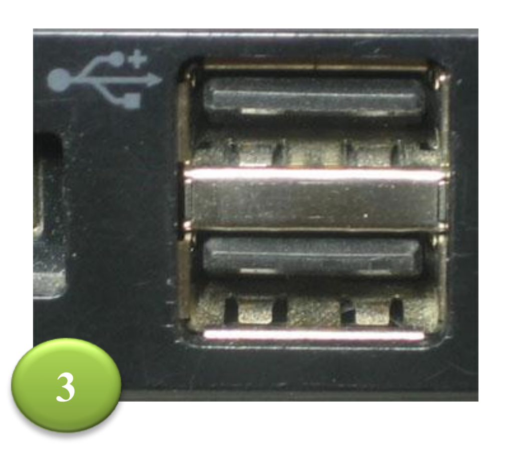

ملحوظه: تعمل البيئه التطويره لاردوينو بصوره اسرع على لينكس اكثر من ويندوز لـذلك انصح اذا كان جهازك المستخدم فى عمليه البرمجه ذو امكانيات متوسطه او ضعيفه من ناحيه Hardware يفضل ان تستخدم احدى نسخ لينكس

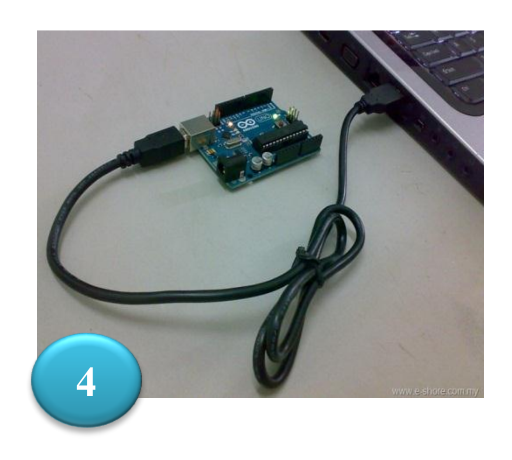
**بعد الانتهاء** من تنصيب Arduino IDE قم بالخطوات التاليه لتوصيل البورده بالكمبيوتر:





- قم بتوصيل بورده اردوينو بالكابل
- ادخل الطرف الاخر فى فتحه USB فى الحاسب الآلى و انتظر قليلا حتى تظهر لك رساله تفيد بأن جهاز الحاسب تعرف على قطعه عتاد جديده Found Arduino Uno Board بعد ظهور تلك الرساله يأتى الوقت لبدأ العمل على البيئه التطويريه





## التعرف على الواجهه الرسوميه لبيئه التطوير



تتكون بيئه التطوير البرمجيه من واجهه بسيطه و تنقسم الى اربعه اجزاء رئيسيه :

**الاول**: شريط القوائم

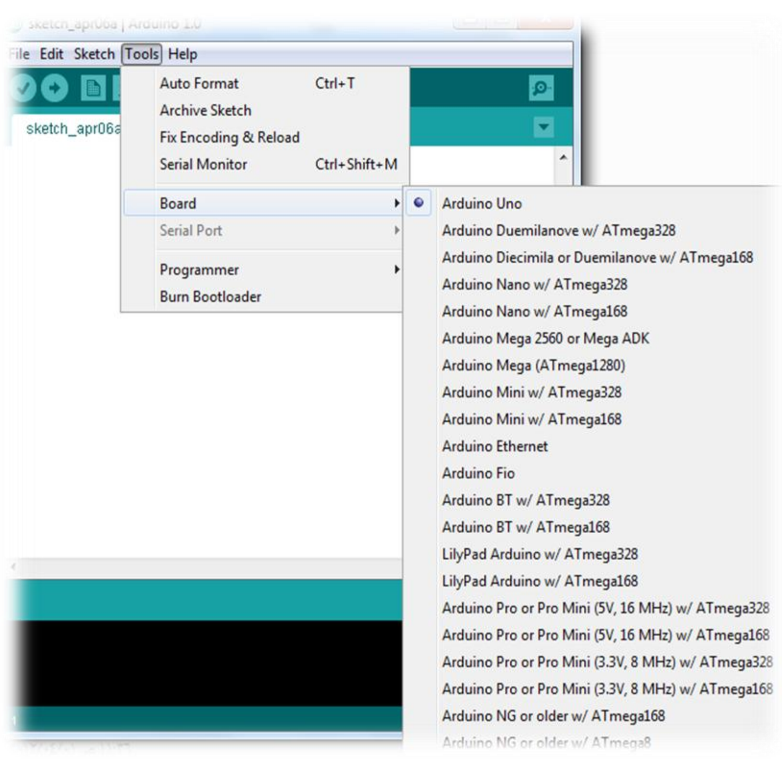
**الثانى**: شريط الاوامر السريعه

**الثالث**: منطقه كتابه الاكواد البرمجيه

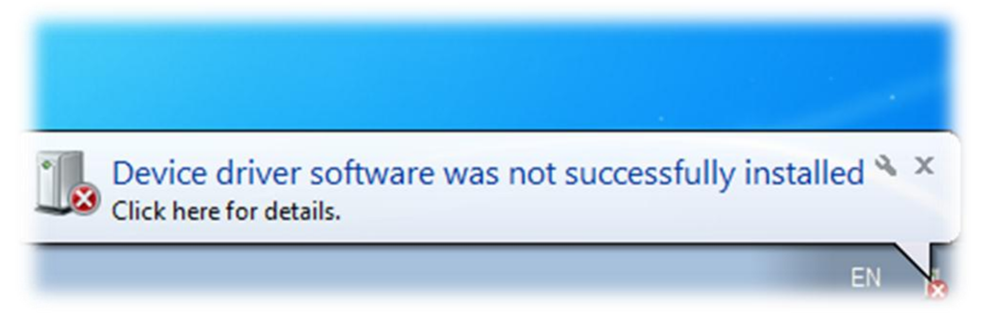
**الرابع**:الجزء الخاص بعرض التنبيهات و الاخطاء البرمجيه فى بيئه التطوير

## تجهيز بيئه التطوير

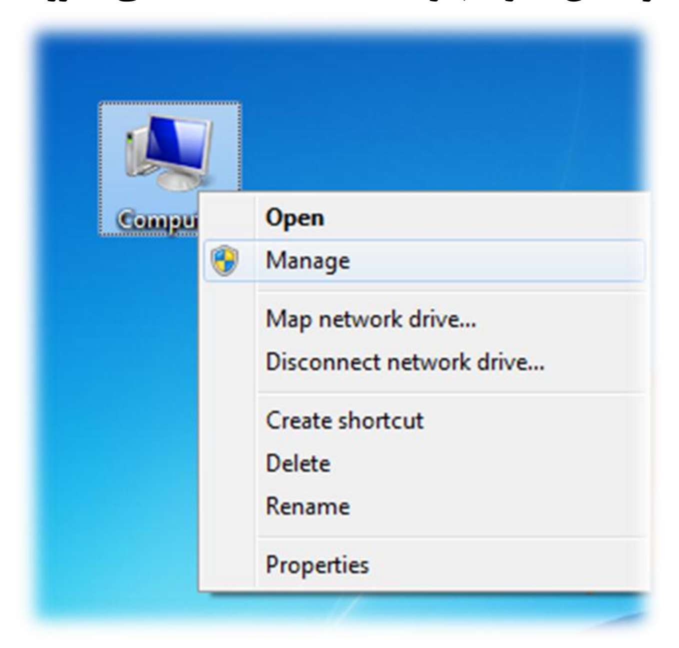
**اول** خطوه لتجهيز البيئه التطويره هى اختيار البورده التى سنتعامل معها و يتم ذلك عن طريق الذهاب الى قائمه Tools و اختيار Boardثم نختار نوع البورده المتوفره لدينا و فى هذه الحاله سنختار Arduino Uno كما فى الصوره التاليه



**ثاني** خطوه تتم فى حاله ظهور هذه الرساله بعد توصيل اردوينو و التى تفيد بأن الجهاز لم يجد تعريفات لكى يستخدم اردوينو (تظهر غالبا مع مستخدمى ويندوز Vista او Windows 7) و يجب ان نقوم بتنصيب التعريفات كالتالى:

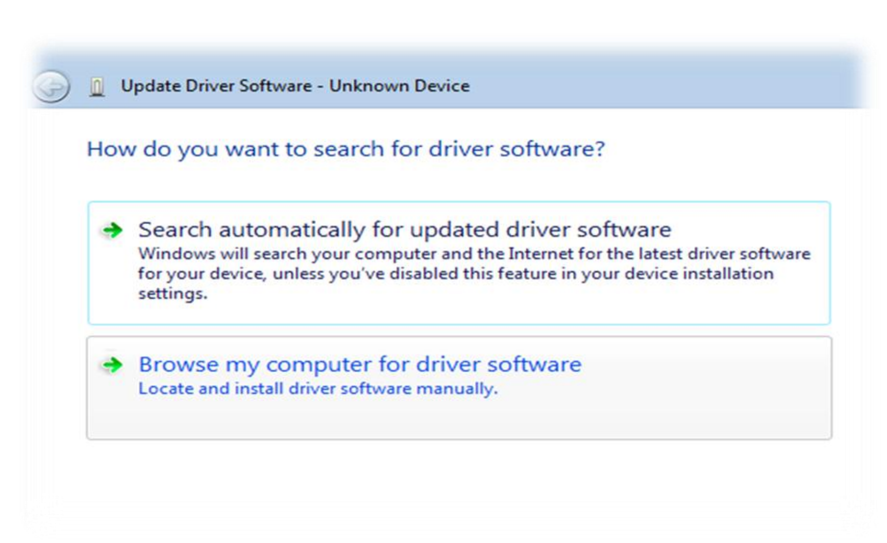


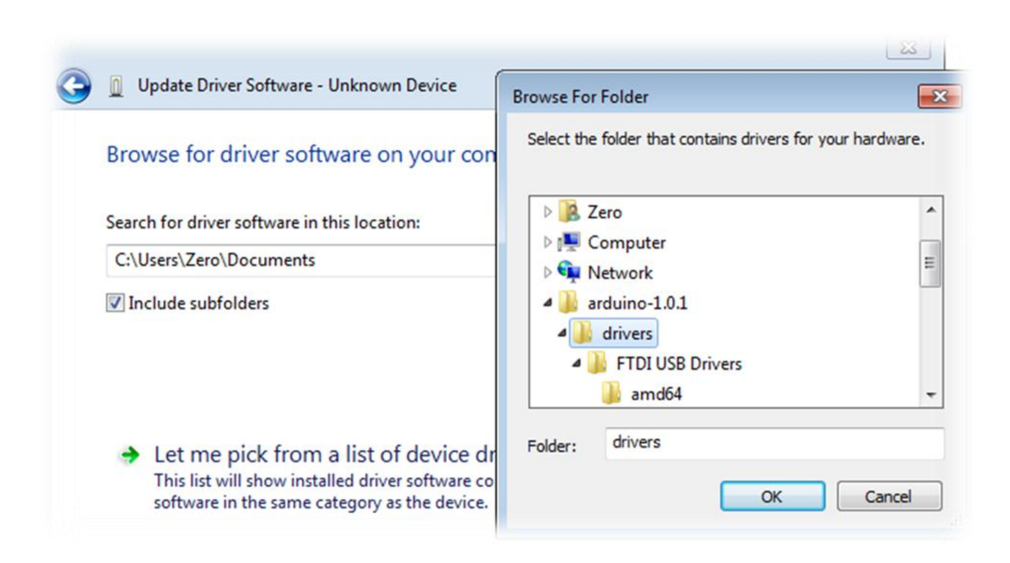

اولا: افتح مدير الاجهزه Device Manager كما فى الصور التاليه



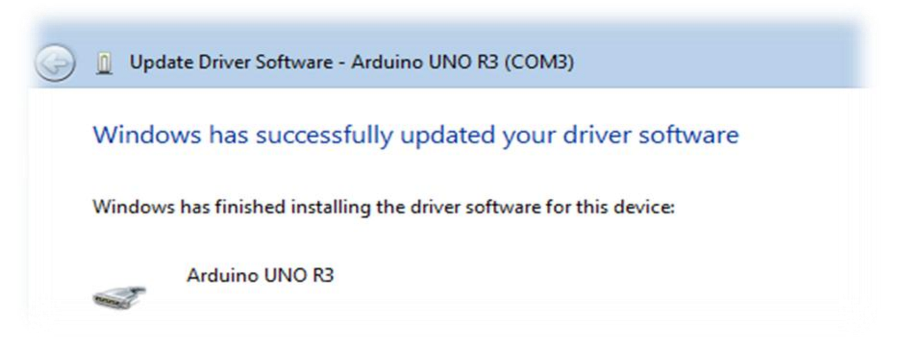
Computer Management
File Action View Help
Computer Management (Local
System Tools
Task Scheduler
Event Viewer
Shared Folders
Performance
Device Manager
Storage
Disk Management
Services and Applications
WIN-0T3MPN6I84S
Batteries
Bluetooth Radios
Computer
Disk drives
Display adapters
DVD/CD-ROM drives
Floppy disk drives
Floppy drive controllers
Human Interface Devices
IDE ATA/ATAPI controllers
Keyboards
Mice and other pointing devices
Monitors
Network adapters
Other devices
Unknown devic
Ports (COM & LPT)
Processors
Sound, video and g
Storage controllers
System devices
Universal Serial Bus
Update Driver Software...
Disable
Uninstall
Scan for hardware changes
**Properties**
Actions
Device
Mo

اختر " تصفح الجهاز لاختيار التعريفات" ثم اختار الفولدر الذى يوجد به برنامج اردوينو



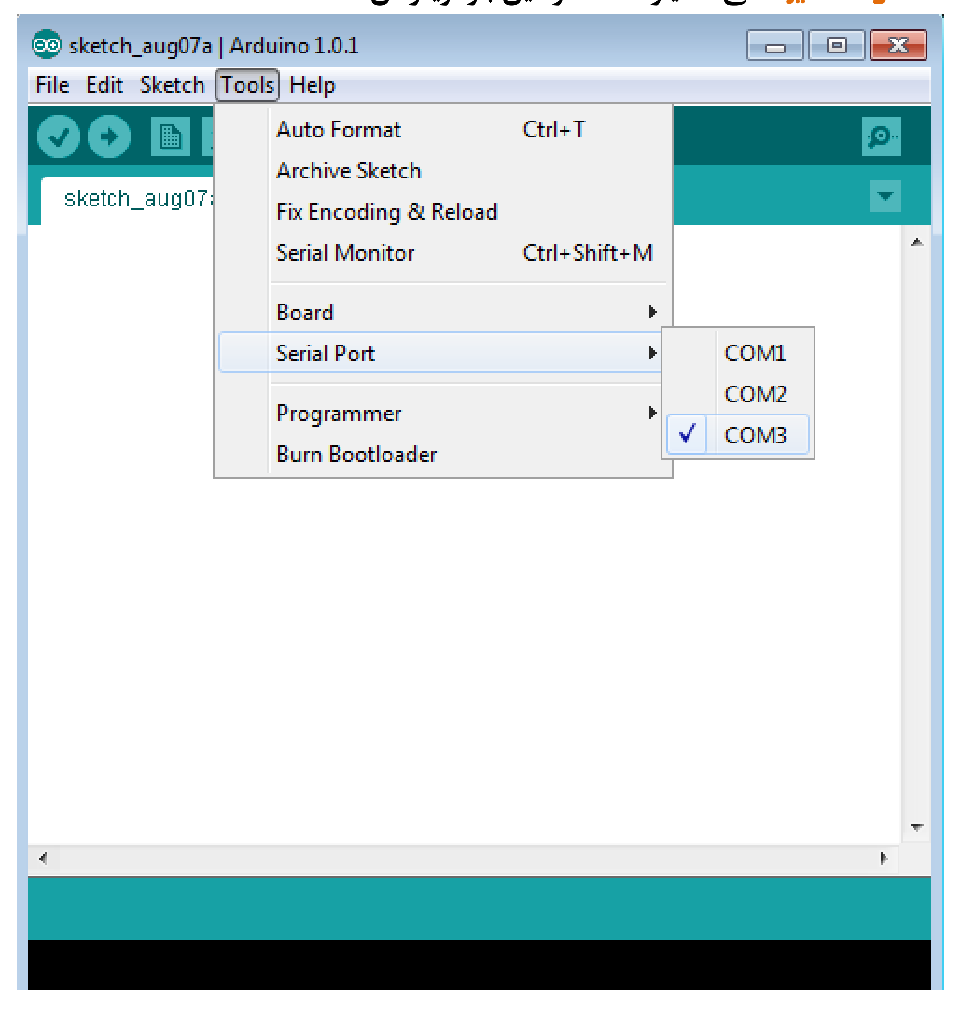



بعدها سوف تظهر رساله تسألك عن "اذا ما كنت ترغب فى تنصيب هذه التعريفات ام لا ؟ " اضغط Okو انتظر قليلا حتى تنتهى عمليه تعريف اردوينو بنجاح كما فى الصوره التاليه

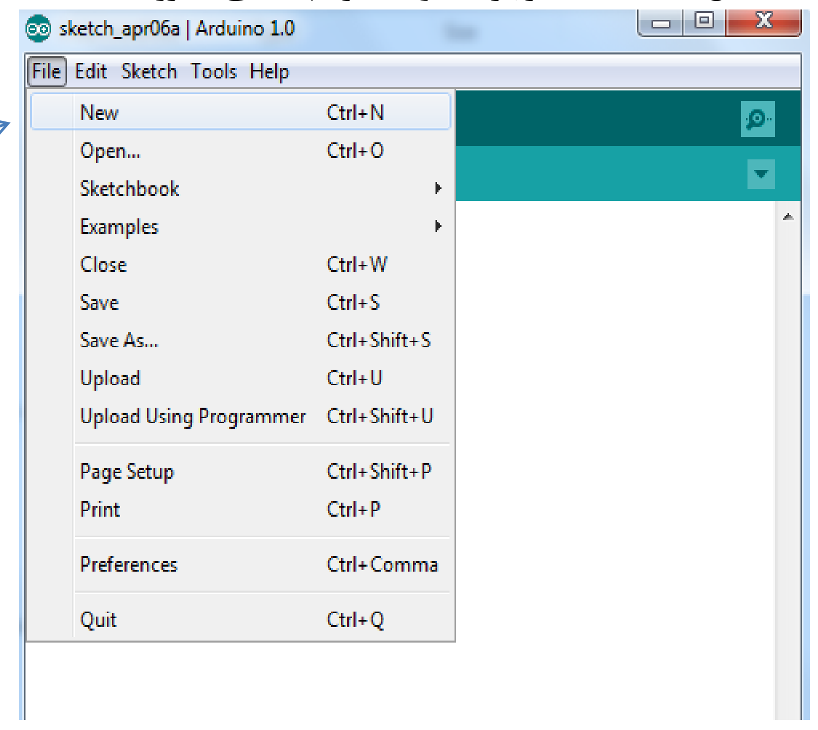



بعد الانتهاء من تنصيب التعريف لاحظ رقم المنفذ المستخدم فى اردوينو من مدير الاجهزه كما فى الصوره التاليه و التى تظهر رقم المنفذ COM3 ، تذكر ذلك الرقم فسوف نحتاجه لاحقاً ( مع ملاحظه ان ذلك الرقم قد يختلف عندك) .

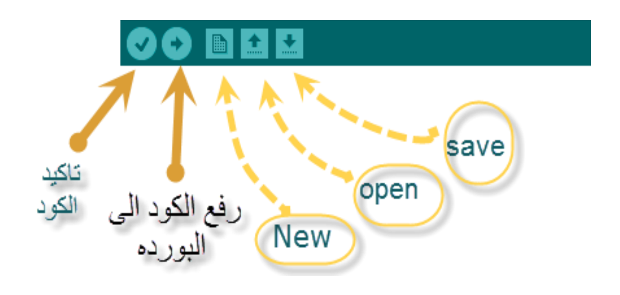
**الخطوه الاخيره** هى اختيار منفذ التوصيل بأردوينو من قائمه Tools – Serial Port



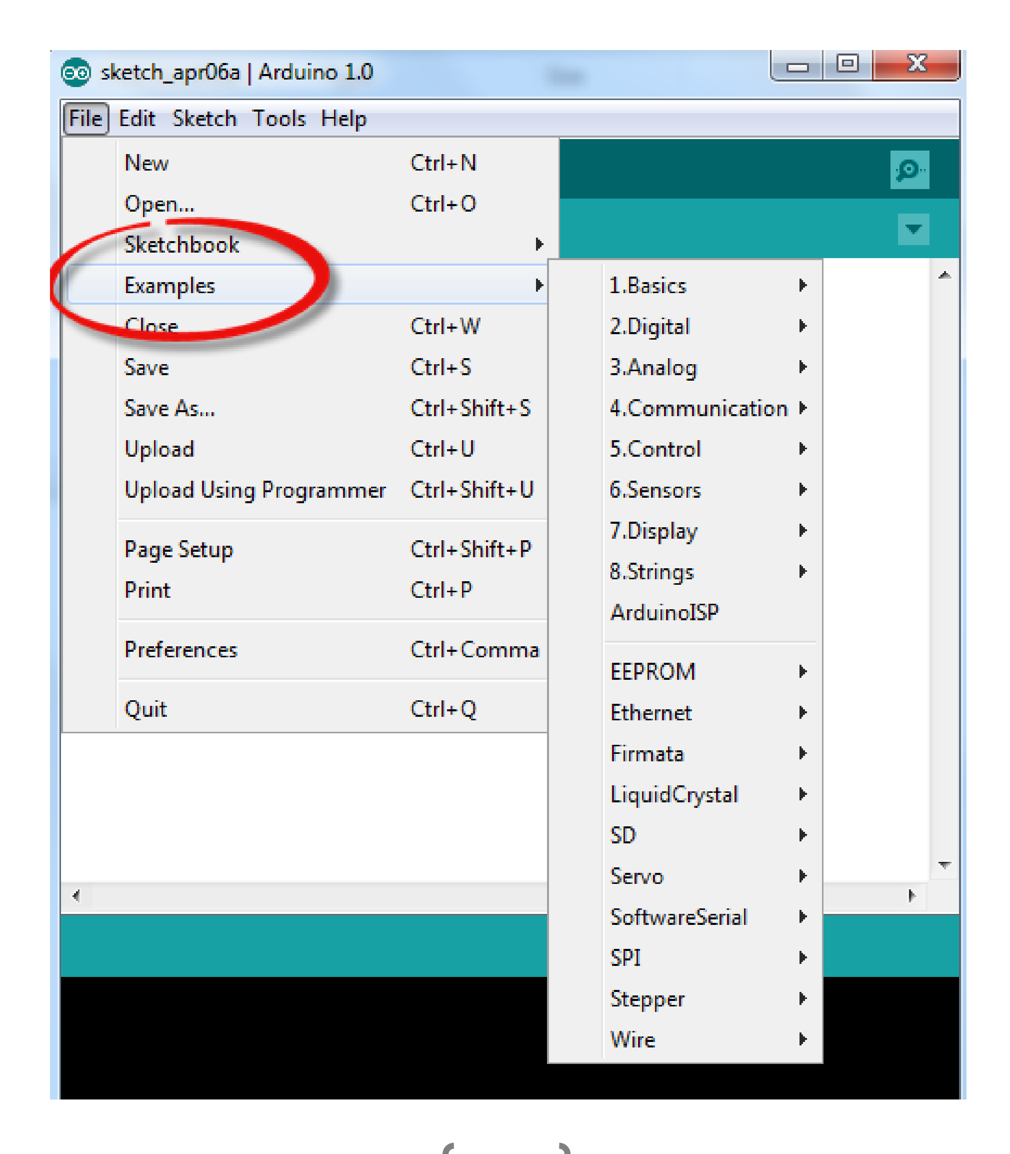
**بعد ذلك** تستطيع البدأ فى كتابه اول برنامج لك على الاردوينو و ذلك عن طريق اختيار New من قائمة File الموجوده بشريط القوائم كما فى الصوره التاليه




**او يمكنك** اختيار New مباشره من شريط الاوامر السريعه



**ايضا** توفر لك البيئه التطويريه مجموعه من الامثله البرمجيه الجاهزه و التى تستطيع الوصول اليها من خلال قائمه Examples الموجوده فى قائمه File الرئيسيه كما فى الصوره التاليه



## ملاحظات شخصيه :

هذه الصفحه مخصصه لكتابه ملاحظاتك الشخصيه عن الفصل الثانى :

اكتب ملاحظاتك
هنا

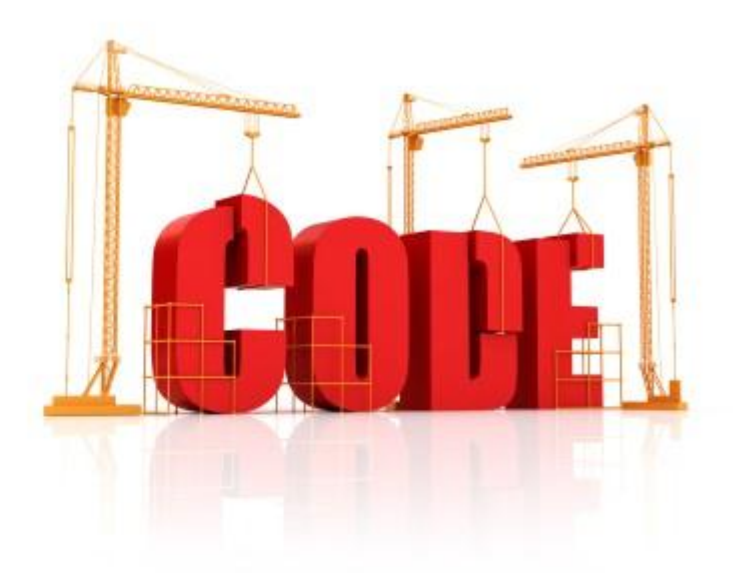
## من أقوال العظماء:

**كُن شَديد التسَامُح مَع من خَالفك الرأي، فإن لم يَكُن رَأيه كل الصواب فلاَ تكُن أنتَ كل الخطأ بتشبُثك برأيك**

فرانسوا ماري أرويه المعروف باسم **فولتير**

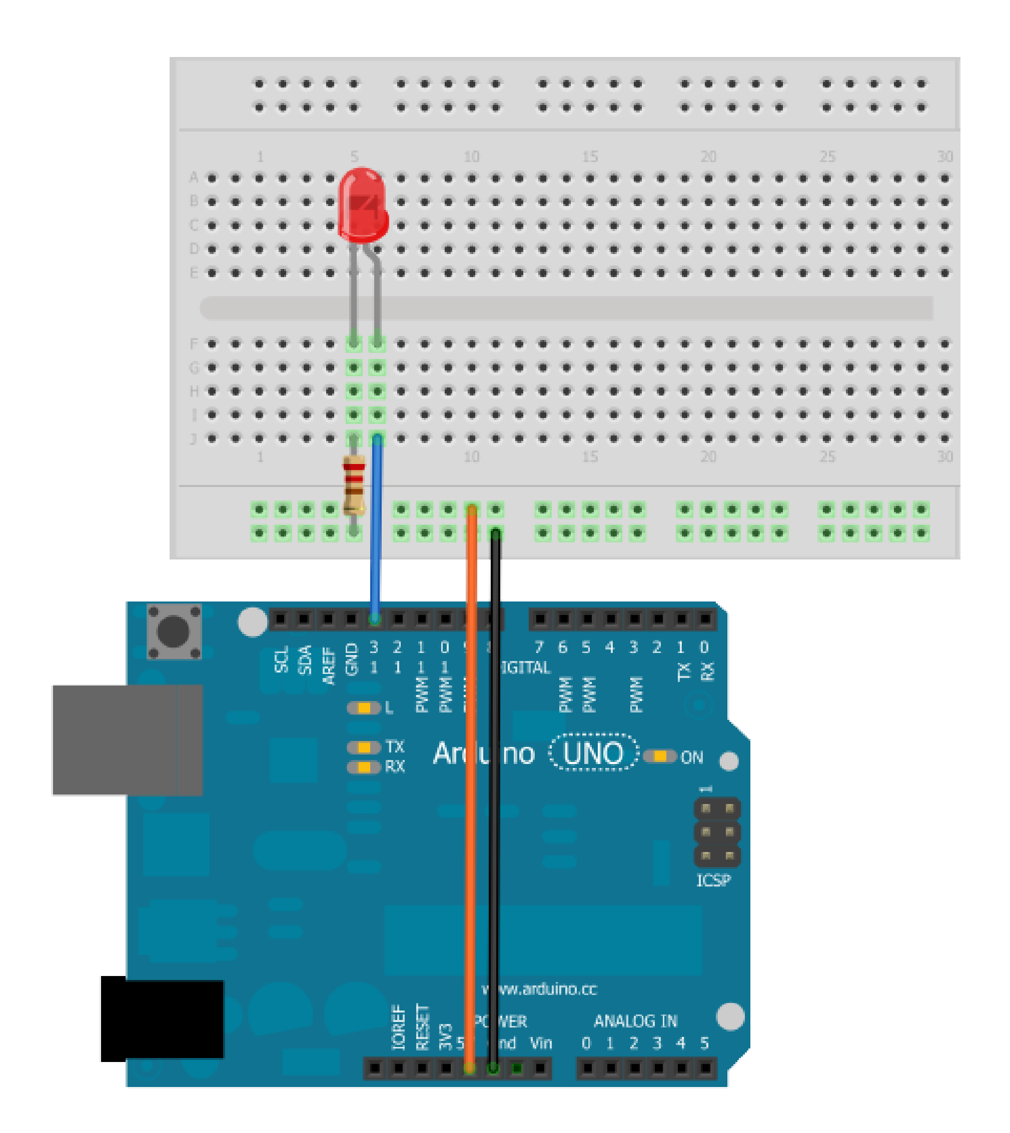
# الفصل الثالث

## لنبدأ العمل مع اردوينو



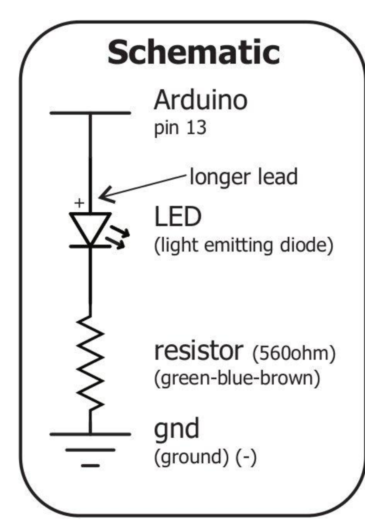

Let's get started

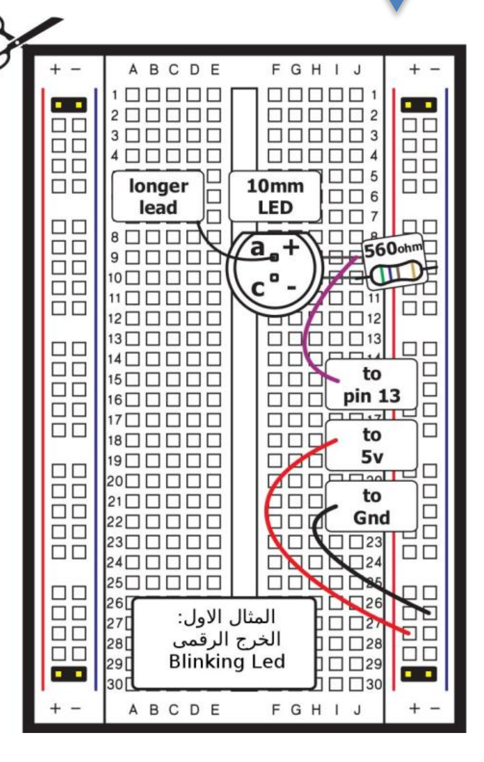
## Example 1: Blinking led المثال الأول: تشغيل دايود ضوئي



## مكونات المثال (1):

- ✓ بورده اردوينو Arduino Uno
- ✓ Breadboard
- ✓ مقاومه كهربيه بقيمه 560 اوم
- ✓ دايود ضوئى قطر 10 ملى
- ✓ جهاز الكمبيوتر
- ✓ ورقه المساعده الموجوده فى هذه الصفحه




**ما هى الورقه المساعده:** هى ورقه يتم طباعتها لترشدك الى كيفيه وضع المكونات الالكترونيه على لوحه التجارب Breadboard

**الخطوه الاولى:** قم بطباعه هذه الصفحه (ملحوظه عند الطباعه لا تقم بتكبير او تصغير حجم الصفحه و استخدم مقياس 100%) ثم قص ورقه المساعده بعد ذلك ضع الورقه فوق Breadboard و ثبتها باستخدام 4 وصلات pin-header ثم ابدأ فى توصيل المكونات الالكترونيه فوقها كم فى الصوره

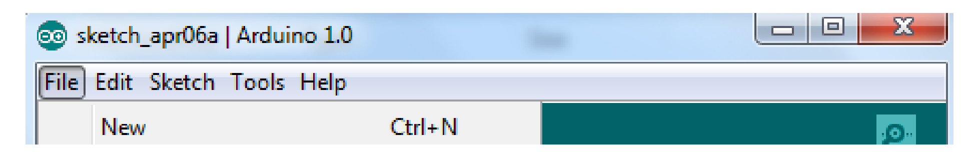
Pin-Header

الورقه المساعده

Breadboard

اردوينو

**الخطوه الثانيه**:من بيئه التطوير Arduino IDE قم بعمل ملف جديد من شريط الاوامر السريعه و اكتب الاوامر البرمجيه التاليه فى الجزء المخصص لكتابه الاكواد



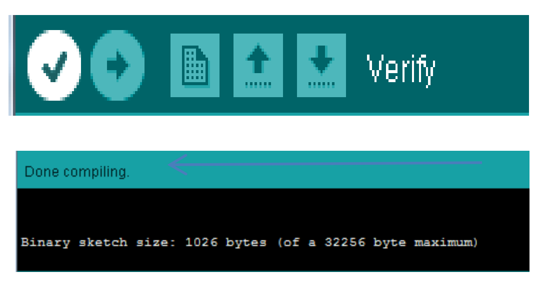
```
// Example 01 : Blinking LED

const int LED = 13;

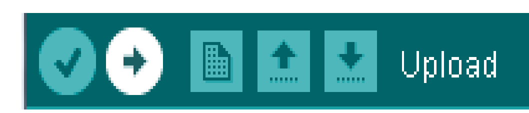
void setup ( )
{
 pinMode(LED, OUTPUT);
}

void loop()
{
 digitalWrite(LED, HIGH);
 delay(1000);

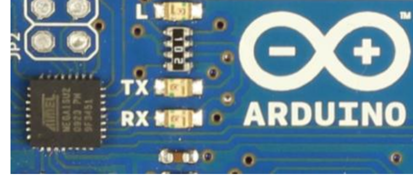
 digitalWrite(LED, LOW);
 delay(1000);
}
```




**الخطوه الثالثه**: بعد ان تنتهى من كتابه الكــود البرمجــى اضـغط علــى زر تأكيــد Verify فــى الشـريط الاوامـر السـريعه فـى اعلى ال IDE ثم انتظر حتى تظهر رساله فى الاسفل تخبرك بأن البرنامج قد انتهى من تجهيز الكـود ليتم رفعه علـى **بـورده الاردوينو**




**تأكــد** ان كابــل الاردوينــو متصــل بالحاسب ثم قم بالضغط على زر " رفع Upload " و ذلك حتـى يتم بـدأ عمليه تحميـل البرنـامج الى المتحكمه الدقيقه و انتظر عـده ثـوانى سـتلاحظ عنـدها ان ال leds الموجــوده علــى البــورده (مكتــوب بجانبها Tx/Rx)بدأت تضئ بسرعه عاليه ثـم تقف قليـل و هـذه الاضائه تعنـى ان البـورده بـدأت تستقبل بيانات البرنامج الذى قمنا بكتابته ، بعد ذلـك يبـدأ البرنـامج و يجعـل الليـد الـذى وضعناه يضئ و ينطفئ بأنتظام كل ثانيه .




**مبروك** ☺ اول مشروع لك بأستخدام اردوينو، و الان سنقوم بتشريح و فهم الكود البرمجى المستخدم فى هذا المثال.



## شرح الكود البرمجي .......

```
const int LED = 13;
```

يقوم هذا الامر بعمل ثابت constant قيمته = 13 و اسمه LED

**سؤال لماذا كتبنا هذا الثابت ؟؟**

نقوم بأستخدام الثوابت حتى تسهل علينا تسميه المخارج و المداخل (Input/Output Pins) للمتحكمه الدقيقه، فمثلا فى هذا البرنامج قمنا بتعريف هذا الثابت حتى نعبر عن المخرج رقم 13 بأن اسمه LED و بذلك لو اردنا ان نأمر المتحكمه فى اى جزء من اجزاء البرنامج لاحقا بأن تشغل الليد الموجود على المخرج رقم 13 او ان تقوم بأطفاء ذلك الليد لن نحتاج عندها لذكر الرقم و انما سنذكر كلمه LED مباشره و ستفهم المتحكمه الدقيقه اننا نقصد بهذا المخرج رقم 13

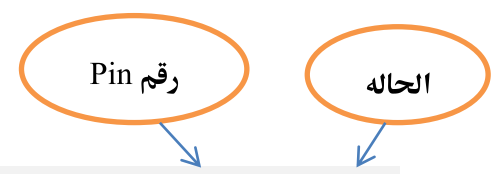
```
Void setup ( )
{
pinMode(LED,OUTPUT);
}
```

**السطر الثانى** من اوامر البرنامج هو تعريف وظيفه ال pin رقم 13 و التى نرمز لها بأسم LED

**في برامج المتحكمات الدقيقه يتكون البرنامج من 3 مراحل و هى كالتالى:**

**المرحله الاولى**: ان تقوم بتعريف اى متغيرات او ثوابت فى البرنامج الذى سوف نقوم بكتابته

**المرحله الثانيه**: تتميز المتحكمات الدقيقه ان جميع ال Digital Pins يمكن ان تستخدم اما كمخرج او مدخل و فى هذا المرحله نقوم بضبط اعدادات المخارج و المداخل و تعريف المتحكمه ان الـ Pin رقم 1 او 12 او 13 مثلا سوف تستخدم كمخرج او ان ال Pin رقم 5 و 6 سوف يستخدمان كمدخل للبيانات و تعتبر الداله `setup ( )` هى المرحله الثانيه.




**يتم** استخدام الامر `pinMode(pin number, state);` لتحديد وظيفه الPin و هل ستعمل كمدخل ام كمخرج حيث يتم كتابه رقم **او** اسم الـ Pin اذا قمنا بتعريف ثابت مخصص سابقا، ثم يتم كتابه حاله التشغيل INPUT **او** OUPUT و لاحظ انه يجب كتابه حروف كلمه (مدخل او مخرج) بصوره capital.

**ايضا** يجب ان تكتب جميع اعدادت المخارج و المداخل بداخل اقواس الداله `setup ( ) {write your configurations here }` على سبيل المثال اذا اردنا ان نخبر المتحكمه ان الـ pin العاشره و الحاديه عشر و الثالثه عشر هم مخارج وان الpin الثانيه هى مدخل فسنقوم بكتابه الاوامر التاليه:

```
void setup ( )
{
pinMode(11,OUTPUT);
pinMode(12,OUTPUT);
pinMode(13,OUTPUT);
pinMode(2,INPUT);
}
```

**لاحظ** : انه يجب بعد ان ننتهى من كل امر يجب ان نضع علامه الفاصله المنقوطه (;) و اذا نسيتها فى اى امر سوف تظهر لك رساله خطأ عندما تقوم بعمل تأكيد verify للبرنامج

**المرحله الثالثه:** فى هذه المرحله نقوم بكتابه البرنامج نفسه و ما نريد من المتحكمه ان تفعله بالظبط و فى حاله البرنامج السابق ستقوم المتحكمه بتشغيل الليد لمده ثانيه ثم تطفئه لمده ثانيه و هكذا مرار و تكرار.

```
void loop( )

{
 digitalWrite(LED, HIGH);
  delay(1000);


 digitalWrite(LED, LOW);

 delay(1000);
}
```

قم بتشغيل LED

ثم انتظر 1000 ملى ثانيه

قم بأطفاء LED

ثم انتظر 1000 ملى ثانيه

**يتم** كتابه برنامج المتحكمه داخل الداله void loop ( ) {your program} ، فى هذا المثال استخدمنا الامر digitalWrite(LED, HIGH); و هو المسؤل عن تطبيق جهد كهربى اما HIGH = 5 Volt او LOW = 0 حيث نقوم بكتابه اسم المخرج اولا ثم الجهد الذى نريد تطبيقه و عندما نريد تشغيل الليد نستخدم HIGH لتطبيق جهد 5 فولت و عند اطفائه نستخدم LOW لتطبيق جهد = صفر.

اما بالنسبه للأمر delay (1000); فهو يستخدم لاخبار المتحكمه كم من الوقت يجب ان تنتظر قبل تنفيذ الامر التالى و بهذا الامر يمكننا ان نتحكم فى كم من الوقت يجب ان يعمل او يطفئ الليد .

و كما شاهدنا فى المثال السابق:

```
digitalWrite(LED, HIGH);
delay(1000);
```

و تلك الاوامر تعنى ان المتحكمه ستطبق جهد يساوى 5 فولت HIGH على المخرج LED ثم تنتظر 1000 ملى ثانى و نلاحظ انه عند كتابه الوقت فى برامج المتحكمات الدقيقه يتم استخدام الملى ثانيه millisecond و التى تعنى جزء من الف من الثانيه و يمكننا استخدام النانو Nano-second و المايكرو ثانيه Micro-Second

بعد ذلك تقوم المتحكمه بتطبيق جهد Low يساوى صفر فولت و الذى يعنى ان الليد سوف ينطفئ ثم يظل منطفئ 1000 ملى ثانيه اخرى = واحد ثانيه .

## ماذا عن اول سطر فى البرنامج؟؟ // Example 1 Blinking led

يحتاج المبرمجين عند كتابه اى برنامج مهما كانت اللغه البرمجيه الى نظام للتعليقات comments و التعليقات هى جمل تساعد المبرمج على فهم و تنظيم الكود البرمجى و لا تحتسب فى اجزاء البرنامج ، بمعنى اخر سيقوم Arduino IDE بتجاهل التعليقات عن نقل البرنامج الى الصوره التنفيذه و نقله و تشغيله على المتحكمه الدقيقه .

يمكنك كتابه التعليقات فى اردوينو فى اى مكان تشاء و يمكن اعتبار اى جمله انها تعليق ان قمت باضافه "//" قبل بدايه السطر.....مثال:

```
// Hello I'm Abdallah
// I'm Using comments
// Arduino is Sooooo funny
```

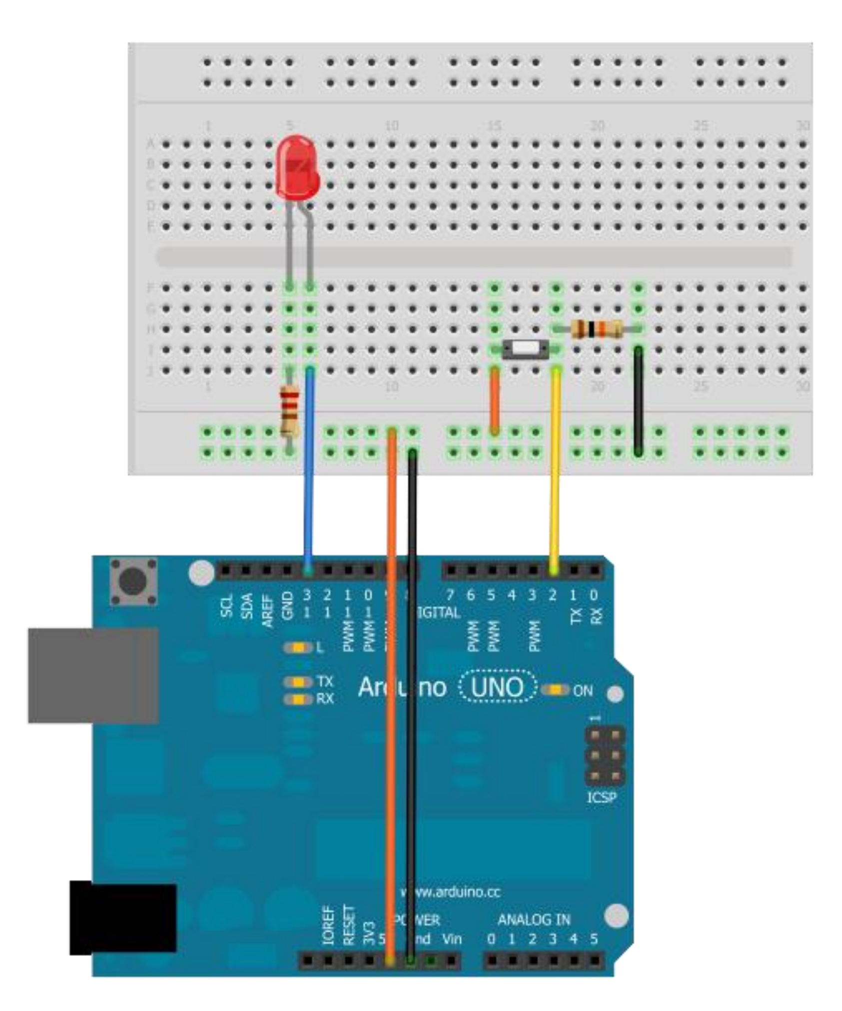
هناك اسلوب اخر لكتابه التعليقات الطويله و هى باستخدام الشرطه المائله و النجمه فى اول الكلام ثم كتابه نجمه و شرطه مائله فى نهايه الكلام .....مثال:

```
/* Hello I'm Abdallah and I Love Arduino, and you
are going to learn The best microcontroller system in
the world (^_^)  */
```

**ملحوظه:** فى الطريقه الاولى لكتابه التعليقات يجب ان تكتب // قبل كل سطر تريد ان يتجاهله بالبرنامج باعتباره تعليق، لكن فى الطريقه الثانيه نكتب /* فى بدايه النص ثم نكتب ما نريد من كلام و نصوص و فى النهايه نختم بـ */، لذلك تستخدم الطريقه الثانيه فى التعليقات الطويله و الطريقه الاولى فى التعليقات القصيره التى غالبا ما تكون سطر واحد.

## المثال الثانى: تطوير للمثال السابق ليعمل فقط عند ضغط مفتاح (سويتش) من نوع Push button

في هذا المثال سنقوم بتطوير الفكره السابقه و سنجعل الليد يعمل فى حاله اذا ضغط المستخدم على زر من نوع push button و هو اشهر انواع السويتشات المستخدمة في الأجهزة الإلكترونية.



## مكونات المثال (2):

- بورده اردوينو Arduino uno
- Breadboard
- Push button
- Led 10 mm
- مقاومه 10 كيلو اوم
- مقاومه560 اوم
- بعض السلوك للتوصيل

**قم بوضع السويتش على لوحه التجارب ثم وصل احدى طرفيه بالمدخل 5v باستخدام سلك على بورده اردوينو و الطرف الاخر بالمقاومه العشره كيلو اوم و من نفس النقطه قم بتوصيل سلك للمدخل رقم2 على اردوينو بورد و بعد ذلك قم بتوصيل طرف المقاومه الثانى بالمدخل الارضى .**

### ما الهدف من هذه الخطوة ؟؟

**تعمل تلك الدائره على توفير دخل رقمي Digital input ، عند ضغط الزر تنطلق نبضه بقيمه 5 فولت و التى تعتبرها المتحكمه الدقيقه اشاره من نوع HIGH و عند ترك الزر ينفصل التيار الكهربى و يصبح الدخل يساوى صفر فولت و الذى تعتبره المتحكمه LOW و سوف نستغل هذا الامر لتعديل المثال الاول لكى يعمل فقط عندما يكون الدخل HIGH**

**الخطوة الثانية :** كتابه الأكواد البرمجية التالية في Arduino IDE

| نصيحه: قم بكتابه الأوامر البرمجيه بنفسك ولا تأخذها نسخ و لصق من الكتاب و ذلك حتي تدرب نفسك على كتابه الكود |
|---|

```
// Example_2_Blinking_with_Switch

const int ledPin = 13;
const int buttonPin = 2;
int val;

void setup ( )

 {
  pinMode(ledPin, OUTPUT);
  pinMode(buttonPin, INPUT);

 }

void loop()

{
  val = digitalRead(buttonPin);

  if (val == HIGH)

     {
  digitalWrite(ledPin, HIGH);
  delay(1000);
  digitalWrite(ledPin, LOW);
  delay(1000);
      }

  else {digitalWrite(ledPin, LOW); } }
```

المرحله الاولى: تعريف المتغيرات و الثوابت

المرحله الثانيه: تعريف و تحديد المداخل و المخارج

المرحله الثالثه: كتابه اوامر البرنامج الرئيسيه

**الخطوة الثالثة:** قم بعمل تأكيد للكود البرمجى عن طريق الضغط على علامه تأكيد Verify و بعد ان تتأكد من ان البرنامج صحيح ولا يحتوى على اخطاء برمجيه و تم تحويله الى الصوره التنفيذيه بنجاح، قم بعد ذلك برفعه للمتحكمه الدقيقه عن طريق زر رفع Upload



## شرح الكود البرمجى ......

**المرحلة الاولى:** قمنا بتعريف متغير اسمه **ledPin** و الذى سيرمز الى Pin 13 و كذلك قمنا بتعريف المتغير المتغير **buttonPin** و الذى سيرمز الى السويتش الموجود على المدخل الرقمى السابع و قمنا بتعريف المتغير **val** و الذى سوف نستخدمه لاحقا لتخزين حاله الدخل على المدخل الرقمى الثانى **pin 2**

```
int ledPin = 13;
int buttunPin = 2;
int val = 0;
```

**المرحلة الثانية:** في هذه المرحلة جعلنا المتحكمة تعامل **pin 13** و التي يرمز لها باسم **ledPin** على أنها مخرج و سنستخدمه لتشغيل الليد الموصل به، ثم قمنا بجعل المتحكمة تعامل المخرج **pin 2** على انه مدخل رقمى وظيفته استقبال الاشارات الرقميه

**HIGH or LOW**

```
void setup()
{
  pinMode(ledPin, OUTPUT);
  pinMode(inPin, INPUT);

}
```

## المرحله الثالثه:

```
val = digitalRead(buttunPin);
```

في هذا الأمر ستقوم المتحكمة بقياس قيمه الجهد على المدخل pin 2 و تسجل القيمة في المتغير val

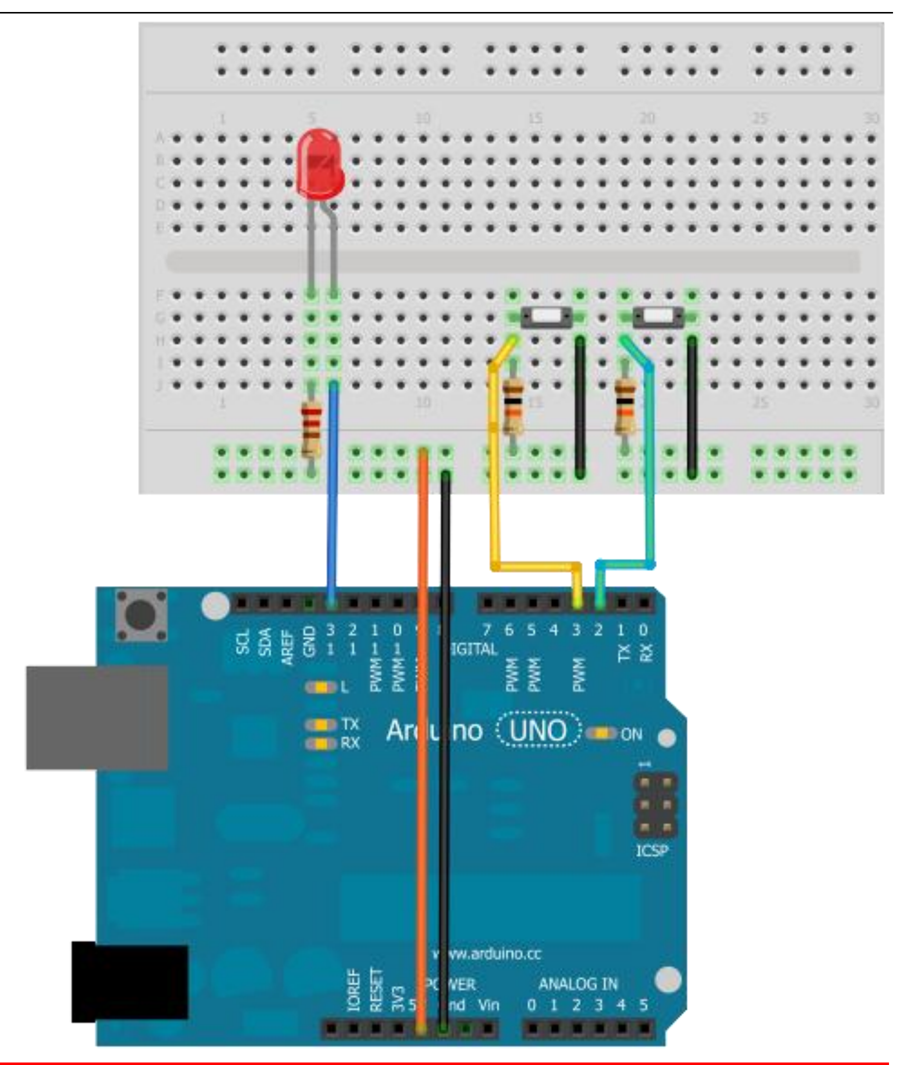
و يتم ذلك باستخدام الأمر digitalRead( ) و اذا كان المستخدم ضغط على السويتش فأن القيمه ستكون 5 فولت = HIGH و ان لم يضغط ستكون القيمه صفر فولت = LOW

```
if (val == HIGH)
   {
digitalWrite(ledPin, HIGH);
delay(1000);
digitalWrite(ledPin, LOW);
delay(1000);
   }
else {digitalWrite(ledPin, LOW); }
```

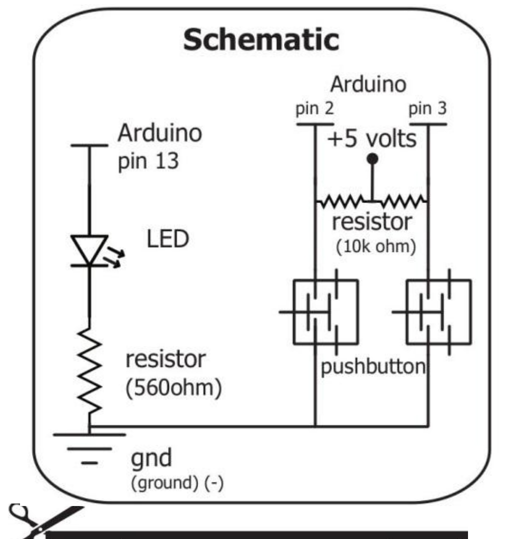
فى هذا الجزء استخدمنا واحد من اشهر الاوامر فى لغات البرمجه و هو الامر ( if....else = اذا .. مادون ذلك )، يستخدم ذلك الامر لعمل مقارنات و جعل المتحكمه تفعل شئ ما اذا حدث شئ اخر محدد و فى هذا المثال استخدمنا الامر if كالتالى :

- ستقيس المتحكمه قيمه الجهد على المدخل السابع و تخزن القميه فى المتغير **val**
- اذا كانت القميه تساوى HIGH و التى تعنى 5 فولت ستقوم المتحكمه بتشغيل الليد لمده ثانيه و اطفائه لمده ثانيه و هكذا باستمرار.
- ما لم تكن القيمه تساوى HIGH و الجهد المطبق على المدخل يساوى صفر ستقوم المتحكمه بتطبيق جهد يساوى صفر على الليد و بذلك لن يعمل الليد و سيظل منطفئ.

## المثال الثالث: تطوير المثال السابق يعمل بـ2 سويتش من نوع Push Button واحد من اجل التشغيل ON و الثانى من اجل الاغلاق OFF



## مكونات المثال(3):

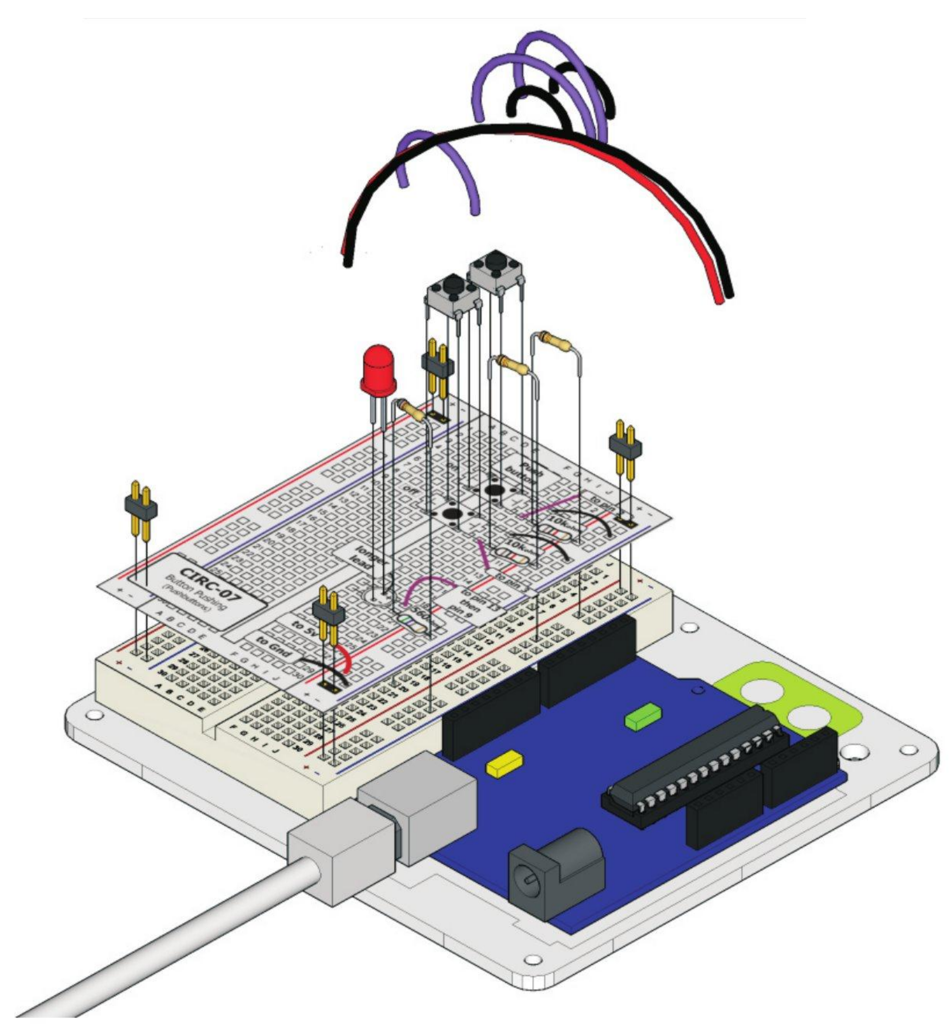



- Arduino uno
- Breadboard
- Led 5 mm
- عدد2 مقاومه قيمه 10 كيلو اوم
- مقاومه واحده قيمه 560 اوم
- عدد 2 سويتش
- اسلاك توصيل
- نسخه مطبوعه من الورقه المساعده

---

في هـذا المثـال سـندمج مـا تعلمنـاه فـي كـلا المثـالين الاول و الثـانى ، سنسـتخدم 2 سـويتش من نـوع **push button** و سيتولى السـويتش الاول عمليـه تشـغيل الليـد و سـيتولى السـويتش الاخر عمليه اطفائه.

**الخطـوه الأولـى :** قـم بطابعـه ورقـه المسـاعده و قصها ثم ثبتها على الــbreadboard باستخدام 4 من الــPin-Headers ثم قم بتوصيل المكوانات الالكترونيه على الورقه المساعده كما في الصوره




**الخطوه الثانيه:** من بيئه التطوير Arduino IDE قم بعمل ملف جديد من شريط الاوامر السريعه و اكتب الاوامر البرمجيه التاليه فى الجزء المخصص لكتابه الاكواد و بعد الانتهاء ارفع البرنامج الى المتحكمه الدقيقه على البورده و جرب الضغط على السويتش الاول و الثانى و شاهد ماذا سيحدث

```
//Example_3_On_Off_with_2_Switchs_1
const int ledPin= 13;
const int inputPin1 = 2;
const int inputPin2 = 3;
void setup ( )
        {
  pinMode(ledPin, OUTPUT);
  pinMode(inputPin1, INPUT);
  pinMode(inputPin2, INPUT);
          }
void loop ( )
        {
  if (digitalRead(inputPin1) == HIGH)
   {digitalWrite(ledPin, LOW); }
else if (digitalRead(inputPin2) == HIGH)
   {digitalWrite(ledPin, HIGH); }
          }
```

تحديد المداخل و المخارج للدائره

اوامر البرنامج الرئيسيه

**في هذا المثال** استخدمنا تعديل بسيط على الأمر if و هو أضافه الأمر **else if** بدلاً من **else** فقط و هو أمر يمكننا من أضافه اكثر من احتمال واحد في جمله **if** .

مثال:

```
If (you like science) {you should read books}
else if (you like sport) {you should play football}
else if (you like art) { you should take art lessons}
else { you should search for a hobby to do in your summer}
```

كما ترى في هذا المثال وضعنا 3 احتمالات و و خيار أخير كتالي :

- **اذا** كنت تحب العلوم **عليك أن** تقرأ بعض الكتب
- **أما اذا** كنت تحب الرياضة **عليك أن** تلعب كره القدم
- **أما اذا** كنت تحب الفن **عليك أن** تتعلم دروس الرسم
- **و اذا لم** يكن لديك هواية **عليك أن** تبحث عن هواية لتشغل بها وقت الإجازة الصيفية

و في البرنامج السابق استخدمنا احتمال أن المستخدم سوف يضغط على السويتش الأول فتقوم المتحكمة الدقيقة بتشغيل الدايود الضوئي و الاحتمال الثاني هو أن يضغط المستخدم السويتش 2 فتقوم المتحكمة بإطفاء الدايود الضوئي ... و هكذا

**و الأن ما رأيك أن نعدل البرنامج قليل ( ^_^ )**

```
//Example_3_On_Off_with_2_Switchs_2
const int ledPin= 13;
const int inputPin1 = 2;
const int inputPin2 = 3;

void setup ( )
{
pinMode(ledPin, OUTPUT);
pinMode(inputPin1, INPUT);
pinMode(inputPin2, INPUT);
}

void loop ( )
{

if (digitalRead(inputPin1) == HIGH)
    {
digitalWrite(ledPin, HIGH);
delay(400);
digitalWrite(ledPin, LOW);
delay(400);
     }

else if (digitalRead(inputPin2) == HIGH)
    {
digitalWrite(ledPin, HIGH);
delay(100);
digitalWrite(ledPin, LOW);
delay(100);
     }

}
```

# مراجعه سريعة على ما تعلمناه فى الامثله السابقه

| | |
|---|---|
| *void setup ( )*<br>*{....................}* | الدالـه المسـؤله عـن ضـبط اعـدادات المتحكمه و تحديد المداخل و المخارج |
| *void loop ( )*<br>*{....................}* | يثم كتابه البرنـامج الرئيسى داخل هذه الداله و يقـوم البرنـامج باعـاده نفسه مره تلـو الاخـرى كلمـا اتم تنفيـذ جميـع الخطوات |
| *int name = value;* | امر يستخدم لتعريف متغير بقيمه معينه |
| *cons int name = value;*<br>**example:** *const int led = 13;* | امر يستخدم لتعريف ثابت غير قابل للتغير يشبه الأمر السابق مع أضافه const |
| *pinMode (pin number, State);*<br>**example1:** pinMode(11,INPUT);<br>**example2:** pinMode(led,OUTPUT); | الامر المسؤل عن تحديد وظيفه كل pin و ضـبطها للعمـل كمـدخل ام كمخـرج INPUT or OUTPUT |
| *digitalWrite (pin number, State);*<br>**example:** *digitalWrite(13,HIGH);* | الامـر المسـؤل عـن اخـراج جهـد علـى احدى الـ pin بصوره رقميه HIGH (1) or LOW(0) |
| *digitalRead (pin number);*<br>**example:** *digitalRead(4);* | الامر المسؤل عن قراءه الجهد المطبق على احدى الـ pin بصـوره رقميه حيـث يقـوم بقيـاس الجهـد (اذا كـان 5 فولـت حتى 3.8 فولت يعتبر HIGH و اذا كان |

| | |
|---|---|
| مـا بـين صـفر حتـى 1.8 فولـت يعتـبر ( LOW | |
| تأخير بزمن محدد : يستعمل هذا الامر فـي تحديـد المـده الزمنيـه للامـر الـذى يسبقه فى التنفيذ | *delay (time);*<br><br>**example1:** *delay(100);*<br>**example2:** *delay(1000);* |
| (اذا حـدث ...او اذا حـدث.. افعل .....)<br>جمله شرطيه تستخدم لتنفيذ امر ما عندما يتحقـق شـرط معـين و يمكـن اسـتخدم الشروط التاليه:<br><br>== يساوي<br>!= لا يساوي<br><= أقل من أو يساوي<br>>= أكبر من أو يساوي<br>< أقل من<br>> أكبر من | if (the condition)<br>{.....what to do}<br><br>else if (another condition)<br>{...what to do}<br><br>else (last condition)<br>{....what to do} |

## اشهر أنواع المتغيرات التي يمكن استخدامها في اردوينو كالتالي:

| النوع | مثال | القميه |
|---|---|---|
| Integers | int led = 13; | أي عدد صحيح من سالب 32768 حتى موجب 32768 |
| Float | float sensor = 12.5; | يستخدم مع الأعداد التي تحتوى كسور |
| character | char name= hello; | حروف نصيه أو كلمات |
| long | long variable = 99999999; | رقم طويل جدا يتراوح بين سالب 2,147,483,648 حتى موجب نفس الرقم |
| byte | byte number= 44; | رقم صغير يتراوح بين (0) إلى 255 |

**لمزيد من المعلومات عن أنواع المتغيرات التي يمكن استخدامها داخل برامج اردوينو قم بزياه الصفحة التالية**

http://www.arduino.cc/en/Reference/VariableDeclaration

## ملاحظات شخصيه :

هذه الصفحه مخصصه لكتابه ملاحظاتك الشخصيه عن الفصل الثالث :

اكتب ملاحظاتك
هنا

## من أقوال العظماء:

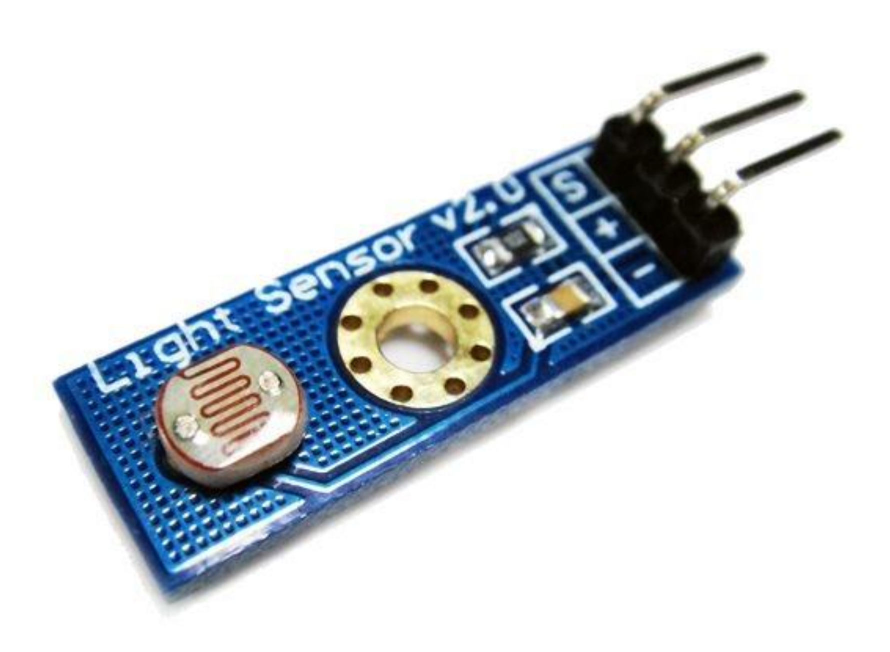
ذو العَقلِ يَشقَى في النّعيمِ بعَقْلِهِ
وَأخو الجَهالَةِ في الشّقاوَةِ يَنعَمُ

أبو الطيب المتنبي

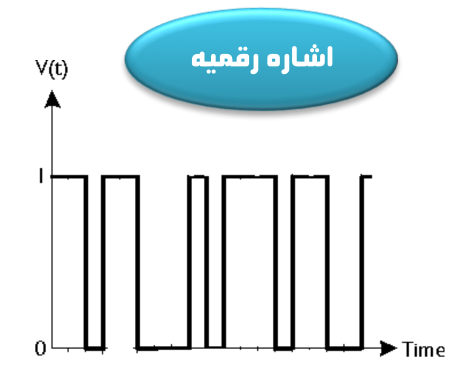
# الفصل الرابع

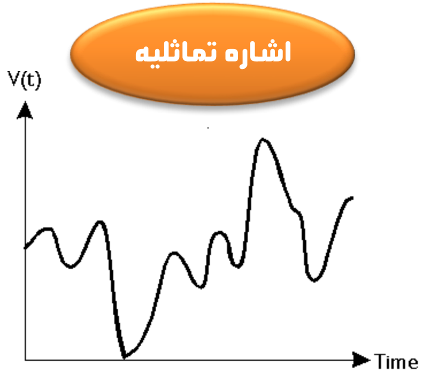
## الدخل و الخرج التماثلى و استخدام المستشعرات

## Analog Inputs/Outputs and Sensors



**يعرف** الدخل او الخرج التماثلى بأنه اى فرق جهد تبدأ قيمته من صفر و قابل للتغير دائما و ليس له حدود قصوى معينه ، على عكس الدخل الرقمى و الذى اما يكون 5 فولت HIGH (1) و اما يكون صفر فولت LOW (0)

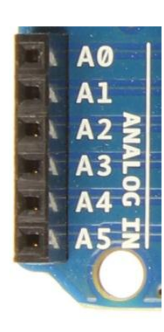



## ما هى أهميه الاشارات التماثليه Analog Signals؟؟

يعبر الدخل التامثلى عن الاشارات المتغيره مثل فرق الجهد الناتج من مستشعر (حساس sensor) معين نتيجه التغير فى عامل معين، على سبيل المثال : حساس للضوء او ما يعرف بأسم المقاومه الضوئيه Photo-Resistor و هى مقاومه كهربيه تتغير قيمتها بتغير مقدار الضوء الذى تتعرض له و اذا قمنا بتوصيل هذا المقاومه على مصدر ثابت للفولت و استخدامنا الAVO-meter لقياس فرق الجهد المطبق عليها .. سنجد ان فرق الجهد سيتغير دائما بتغير مقدار الضوء الساقط عليها

**و يمكننا ان** نستغلال تلك الظاهره فى قياس اى عامل بيئى معين باستخدام مستشعرات مناسبه و التى تقيس ذلك العامل البيئى و تحوله الى اشاره كهربيه تماثليه قابله للقياس مثـل: الحـراره، الضـوء، الرطوبـه، سـرعه الريـاح، القـوه، العـزم، التسـارع، المجـال المغناطيسى... الخ




تحتوى شريحه Atmega 328-p المستخدمه فى بورده Arduino Uno على 6 مداخل تستطيع قياس مقدار فرق الجهد الكهربي المتغير (دخل تماثلي) و هي مرقمه من A0 حتى A5 و يمكنها قياس فرق جهد بدقه تصل الى 4.8 مللى فولت و هذا يعنى تقريبا خمسه اجزاء من الالف جزء من واحد فولت و هى دقه عاليه تكفى للكثير من التطبيقات الدقيقه

**فى هذا الفصل** سنتعرف على بعض اشهر المستشعرات مثل المقاومه الضوئيه و مستشعر الحراره مثل TMP36 أو LM35 هو عبارة عن ترانزستور خاص يتأثر بالحرار و يعطى تغير فى الجهد الكهربى (الفولت) متناسب مع التغير في درجة الحرارة .. حيث يعطي 10 مللى فولت (عشره أجزاء من الف فولت) لكل 0.1 درجة مئوية.... كما سنتعرف على الدخل و الخرج التماثي و كيفيه استخدام المستشعرات مع اردوينو



## كيف تولد الحساسات الإشارة التماثلية ؟؟

لنأخذ حساس الحرارة كمثال: حساس الحرارة مثل TMP36, TMP35, LM35 ما هو الا ترانزستور ذا حساسيه عالية للتغير في الحرارة و من المعروف أن الترانزستور يصنع من ماده السيلكون التي تتغير مدى قابليتها لتوصيل الكهرباء بتغير الحرارة حيث يزداد التيار الكهربي المار فيها بزياده الحرارة و يمكننا إجراء تجربه بسيطة لنتعرف اكثر على طريقه عمل ذلك الحساس.

يحتوى الحساس على ثلاث ارجل وهى كالتالي:

1- منفذ الدخل و يتم توصيل جهد ثابت بين 2.2 فولت حتى 5.5 فولت
2- منفذ الخرج و هو المنفذ الذى نحصل منه على قراءة الحساس
3- منفذ الأرضي و يتم توصيله بأى نقطه ارضى GND





سنحتاج لهذه التجربة المكونات التالية:

1- عدد 2 بطاريه مقاس AAA بقيمه 1.5 فولت
2- عدد 1 حامل بطاريات لتوصيل 2 بطاريه
3- جهاز قياس متعدد الاستخدامات AVO Meter
4- حساس حرارى من أي الأنواع التالية : TMP 35, TMP36, LM 35




احضر 2 بطاريه بقياس AAA بقيمه 1.5فولت و ضعهم في علبه حامل البطاريات لتحصل على قيمه كليه تساوى 3 فولت ثم وصل الطرف الموجب لحامل البطارية (سلك لونه احمر) بالطرف الأول من ناحيه اليسار للحساس الحرارى (طرف الدخل الثابت)

ثم قم بتوصيل الطرف الأرضي لحامل البطارية (سلك لونه اسود) بالطرف الأخير للحساس الحرارى ، بعد ذلك قم بضبط الـ AVO على وضع قياس الفولت و وصل الطرف الأحمر بالمخرج للحساس الكهربي و الطرف الأسود للطرف الأرضي للحساس الكهربي



لاحظ قراءة فرق الجهد الظاهر على الـ AVO meter ستجد أنها 0.76 فولت تقريبا

**الأن قم** بوضع يدك على الحساس و اضغط عليه قليلا (تعمل تلك الحركة على رفع درجه حراره الحساس و من المعلوم أن درجه حراره الجسم تساوى 37 درجه مئوية تقريبا و هي اعلى من درجه حراره جو الغرفة )

ثـــم لاحـــظ الـــتغير في قـــراءة الفولـــت النـــاتج كمـــا في الصـــورة التاليـــة:

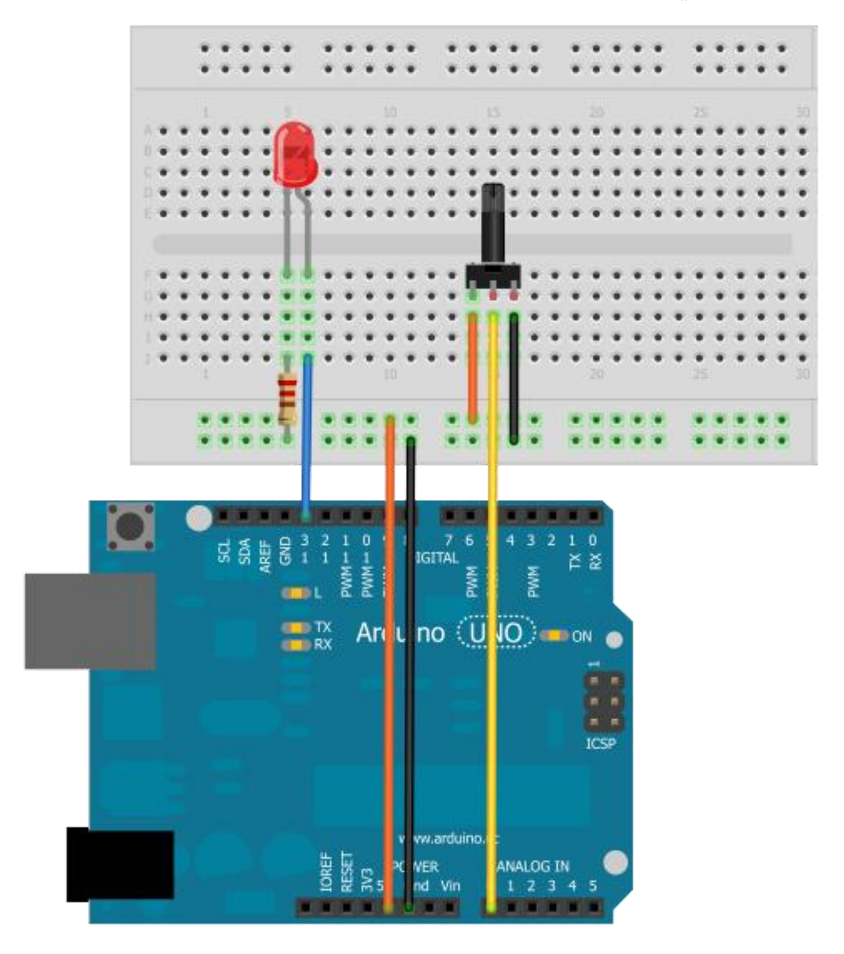



**سنجد أن** قميه الفولت قد ازدادت بسبب حراره اليد

تعمل جميع الحساسات بأنواعها المختلفة بنفس المبدأ حيث يؤدى التغير في عامل بيئي معين إلى تغير المقاومة الكهربية الداخلية مما يـؤدى إلى التغير في فرق الجهد الخارج مـن الحسـاس مما يولـد أشـاره تماثليـه مـتغيره يمكـن قياسـها ، و الأن لنبـدأ مـع امثلـه استخدام الحساسات مع اردوينو

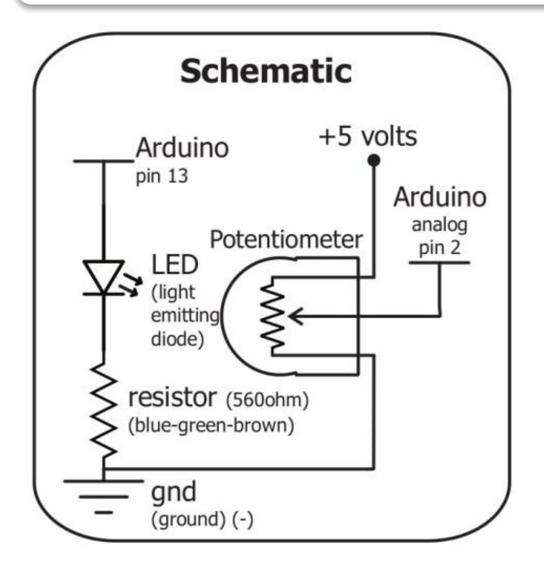
# المثال الرابع: استخدام مقاومه متغيره للتحكم فى سرعه اضائه دايود ضوئى Potentiometer

في هذا المثال سنستخدم مقاومه متغيره للحصول على فرق جهد يمكن تغيره (دخل تماثلي analog input) و سنجعل الدايود الضوئى يضيئ و ينطفئ بسرعه تعتمد على قيمه الدخل التماثلي

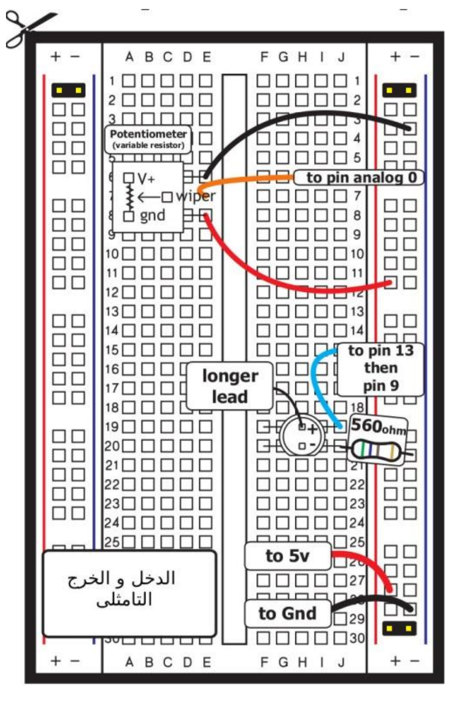


## مكونات المثال (4):

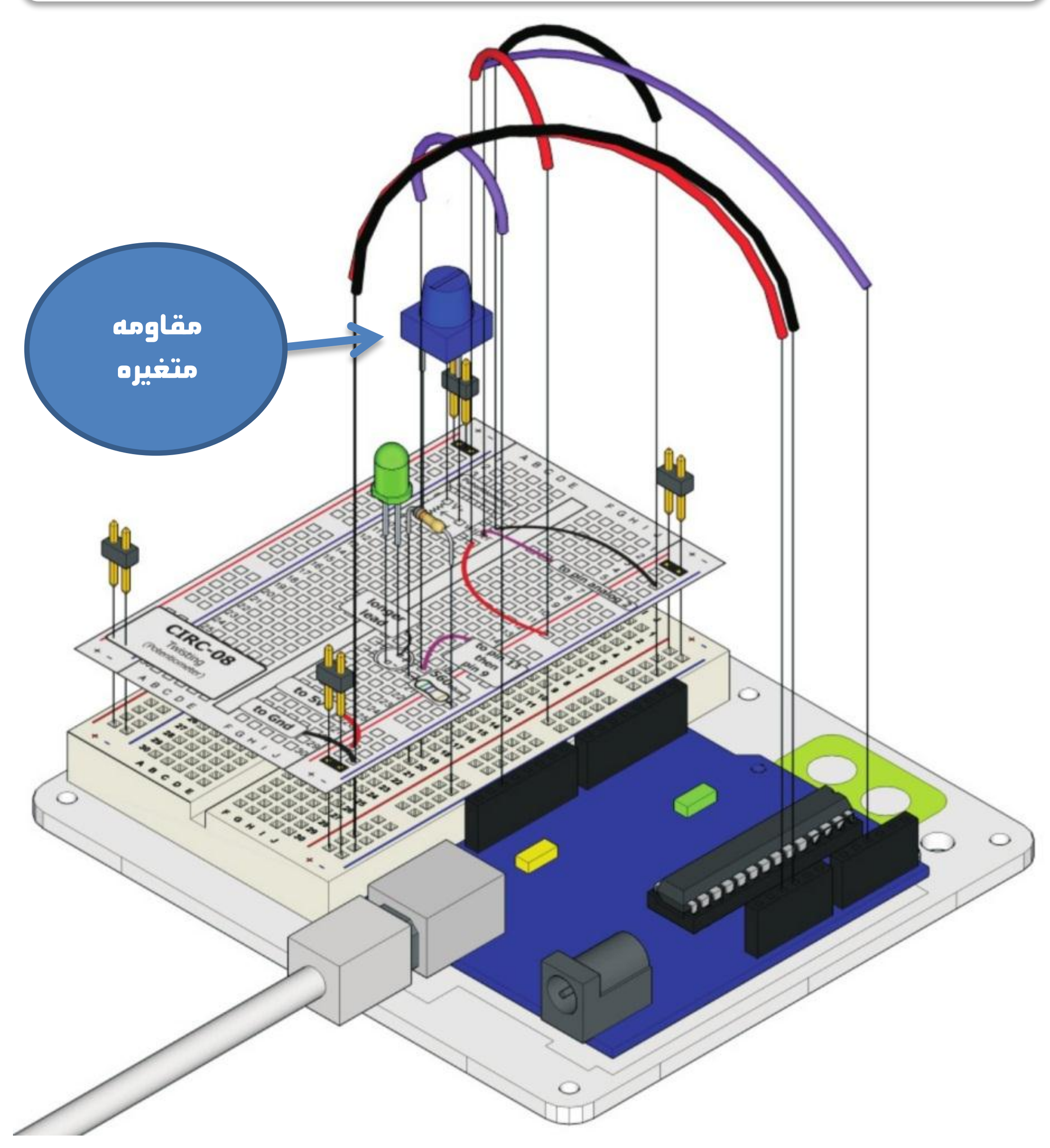
- ✓ بورده اردوينو Arduino Uno
- ✓ لوحه تجارب Breadboard
- ✓ دايود ضوئى led 5mm
- ✓ مقاومه 560 اوم
- ✓ مقاومه متغيره 3 اطراف 10 كيلو اوم
- ✓ أسلاك توصيل
- ✓ ورقه المساعدة المطبوعة




**قــم** بطباعــه الورقــه المســاعده ثــم وصــل المكونات الالكترونيه على لوحه التجارب




**ملحوظه:** لا يشترط ان تستخدم مقاومه متغيره صغيره الحجـم مثـل التـى فـى الصـوره و لـك الحريه فـى استخدام اى مقاومه متغيره لها 3 اطراف بأى حجم و اى قيمه مثل 100 كيلـو اوم او 200 كيلـو اوم، **ايضـا** جميع المقاومات المتغيره ذات 3 اطراف يكـون الطرف الاوسط هو الطرف المتغير و الذى سنوصله بالمدخل التمـاثلى A0 علـى بـورده اردوينـو كمـا فـى الصوره التاليه




**بعد الانتهاء** من توصيل المكونات الالكترونيه على لوحه التجارب و بورده اردوينو قم بعمل ملف جديد و اكتب الاوامر البرمجيه التاليه ثم قم برفع البرنامج الي بورده اردوينو

```
//Example_4_Variable_Resistor_sensor
const int sensorPin = A0;
const int ledPin = 13;
int sensorValue;

void setup ( )
{
  pinMode(ledPin, OUTPUT);
}
void loop ( )
{
  sensorValue = analogRead(sensorPin);
  digitalWrite(ledPin, HIGH);
  delay(sensorValue);
  digitalWrite(ledPin, LOW);
  delay(sensorValue);
}
```

متغير لتخزين قيمه الجهد التماثلى

## شرح الكود البرمجى

**فى هذا المثال** سنتعرف على واحده من اهم الدوال البرمجيه الموجوده فى لغه اردوينو **analogRead(pin number)** تقوم هذه الداله بقرأه فرق الجهد بصوره تماثليه و تستطيع المتحكمه الدقيقه ان تقيس فرق جهد من 4.8 مللى فولت حتى 5 فولت تقريبا و تقوم بتحويل الاشاره التماثليه الى قيمه رقميه من صفر الى 1024 و تسمى هذه العمليه باسم تحويل الاشاره من تماثليه الى رقميه Analog to digital converting

**مثلا** اذا كان الجهد الداخل الى A0 يساوى القيم التاليه:

**4.8 مللى فولت** = 1 **رقمى**
**48 مللى فولت** = 10 **رقمى**
**480 مللى فولت** = 100 **رقمى**
**واحد فولت** = 208.33 **رقمى**
**2 فولت** = 416.66 **رقمى**
**5 فولت** = 1024 **رقمى**

و يستخدم الامر التالى فى تخزين قيمه القرأه الناتجه داخل المتغير sensorVlaue

```
sensorValue = analogRead(sensorPin);
```

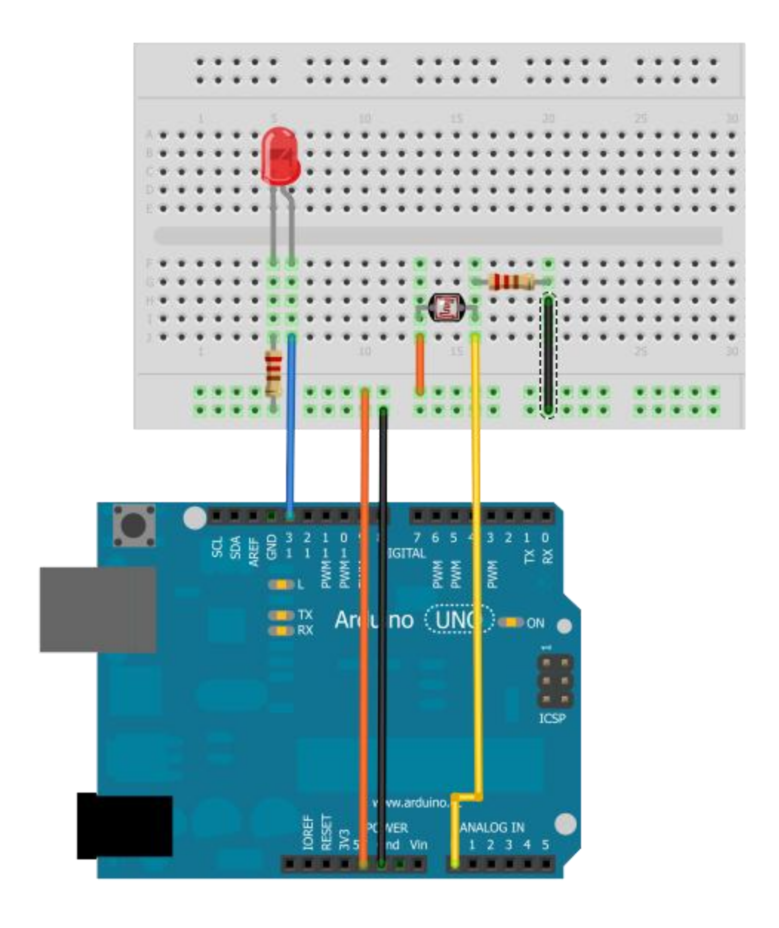
**بعد ذلك** تقوم المتحكمه بتشغيل الدايود الضوئى فتره زمنيه تساوى قيمه هذا المتغير و تقوم بأطفائه بنفس الزمن مثل المثال الاول blinking led ولكن الزمن فى هذا المثال متغير تبعا لقرأه الجهد التماثلى على المدخل A0.

في هذا المثال استخدمنا مقاومه متغيره يمكننا ان نغير قيمتها عن طريف لف (تدوير) عمود الدوران الخاص بها و بذلك نحصل على اشاره تماثليه (تسمى تلك العمليه تغير متحكم به من الانسان) ، فى المثال القادم سنتعلم نوع جديد من المقاومات المتغيره و هى المقاومه الضوئيه التى تتغير قيمتها بتغير كثافه الضوء الساقط عليها و يسمى هذا تغير بيئى و سوف نستغل ظاهره التغير فى المقاومه بتغير الضوء لنحول المقاومه الى مستشعر للضوء light sensor

# المثال الخـامس: المقاومـه الضـويه (حسـاس الضـوء التماثلي) Photo resistor as light sensor

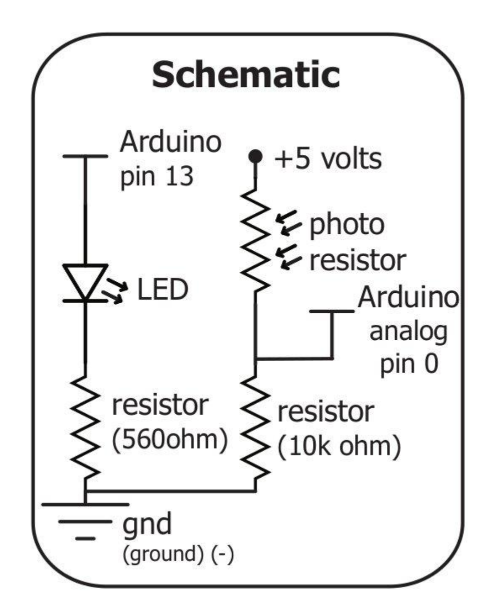
---

**تصنع** المقاومات الضوئية عاده من نـوع خاص من السيلكون المُعالج، و يعتبر السيلكون المعالج ماده شبهه موصله للكهرباء و تتأثر مدى توصيليته للكهرباء بالضـوء لـذلك يستخدم في الحساسات الضوئية و الخلايا الشمسية .



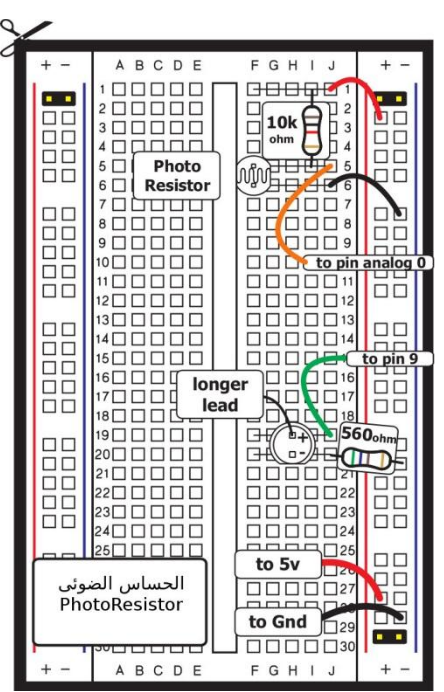
## مكونات المثال (5):

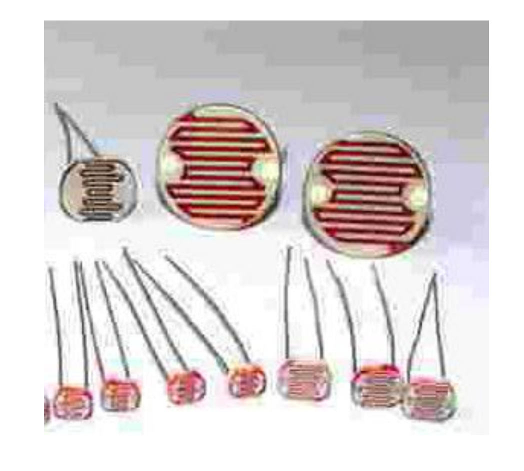
- ✓ بورده اردوينو Arduino Uno
- ✓ لوحه تجارب Breadboard
- ✓ دايود ضوئى led 5mm
- ✓ مقاومه 560 اوم
- ✓ مقاومه ضوئيه
- ✓ اسلاك توصيل
- ✓ ورقه المساعده المطبوعه

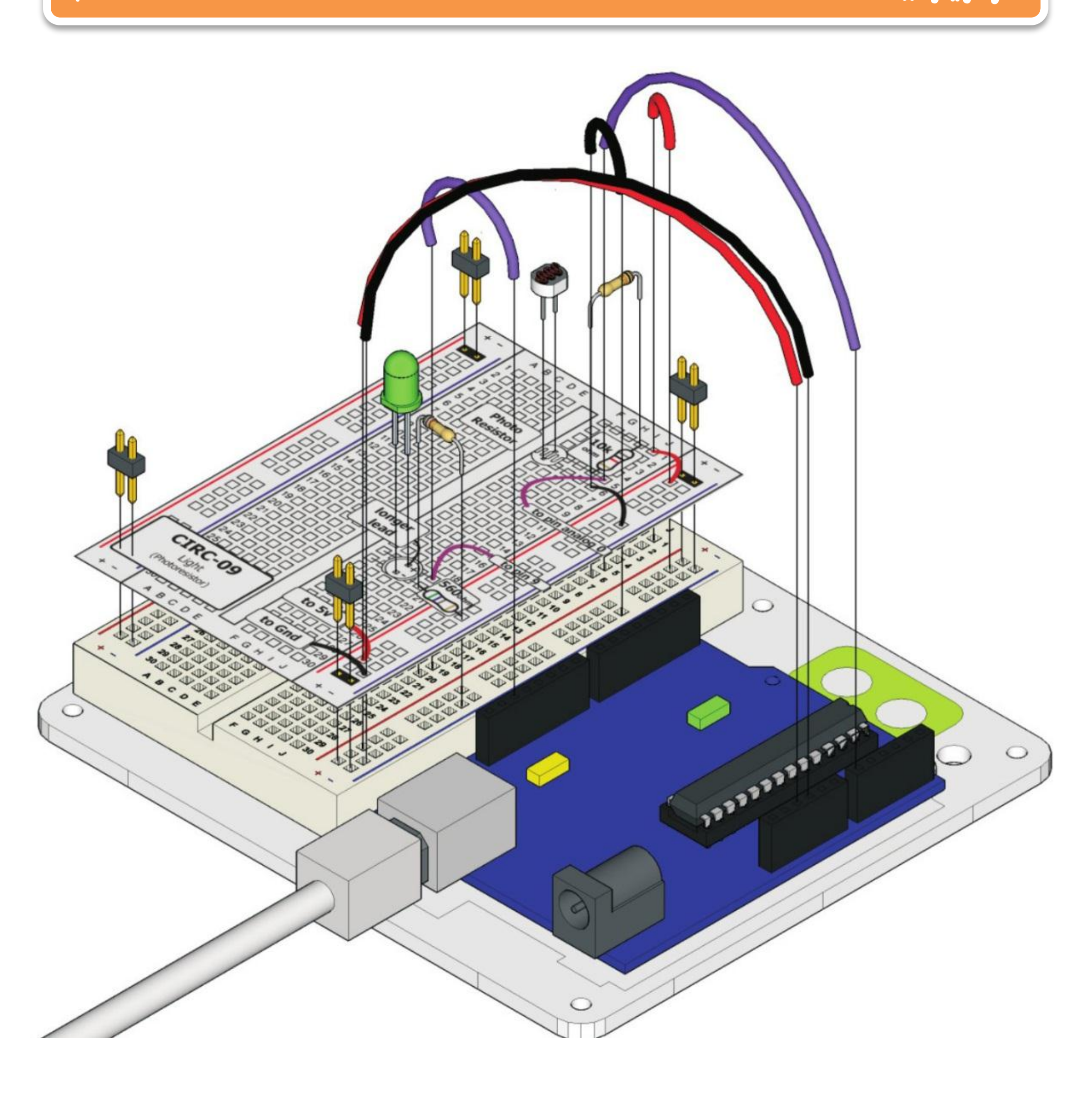



---

تتوفر المقاومات الضوئية في احجام مختلفه تبعا لمقدار حساسيتها للضوء و يمكن استخدام اى حجـــــــــــم متـــــــــــــــوفر




CIRC-09
Light
(Photoresistor)
to 5v
to Gnd
longer lead
Photo Resistor
10k
560
to pin analog 0
to pin 9

```
//Example_5_Light_Sensor_led
const int lightPin = A0;
const int ledPin = 9;
int lightLevel;

void setup ( )
{
  pinMode(ledPin, OUTPUT);
}

void loop ( )
{
 lightLevel = analogRead(lightPin);
 lightLevel = map(lightLevel, 0, 900, 0, 255);
 lightLevel = constrain(lightLevel, 0, 255);
 analogWrite(ledPin, lightLevel);
}
```

**قم بتنفيذ البرنامج** و رفعه على بورده المتحكمه الدقيقه ثم شاهد ماذا سيحدث للدايود الضوئى عند تعريض المقاومه الضوئيه للضوء، ثم حاول ان تقلل مقدار الضوء الساقط على المقاومه الضوئيه عن طريق تغطيتها بيدك و شاهد ماذا سيحدث الى الدايود الضوئى.

قيمه الخرج
رقم المخرج




الأن سنتعرف analogWrite (pin number, value); و هى الداله المسؤله عن توليد خرج تماثلى analog Output و يمكن تطبيقها على المخارج التى تدعم خاصيه التعديل الرقمي المعتمد على عرض النبضة(Pulse-Width modulation) و هى المخارج التى يكتب بجانبها علامه (~) . و هى 6 مخارج كالتالي: **3,5,6,9,10,11**

## ما هي خاصيه التعديل الرقمي PWM ؟؟

خاصيه تمكنك من انتاج فرق جهد على هيئه موجه قابله للتغير عن طريق اشاره رقميه، و بذلك تستطيع أن تحول الأوامر الرقمية إلى موجه تماثليه و يمكن استغلال تلك الخاصية في توليد جهد متغير قيمته بين صفر فولت حتى 5 فولت عن طريق استخدام قيم رقميه من 0 حتى 255 مثلا:

255 = 5 volt
128 = 2.5 volt
0 =0 volt

## كيف نستفيد من هذه الخاصيه؟؟

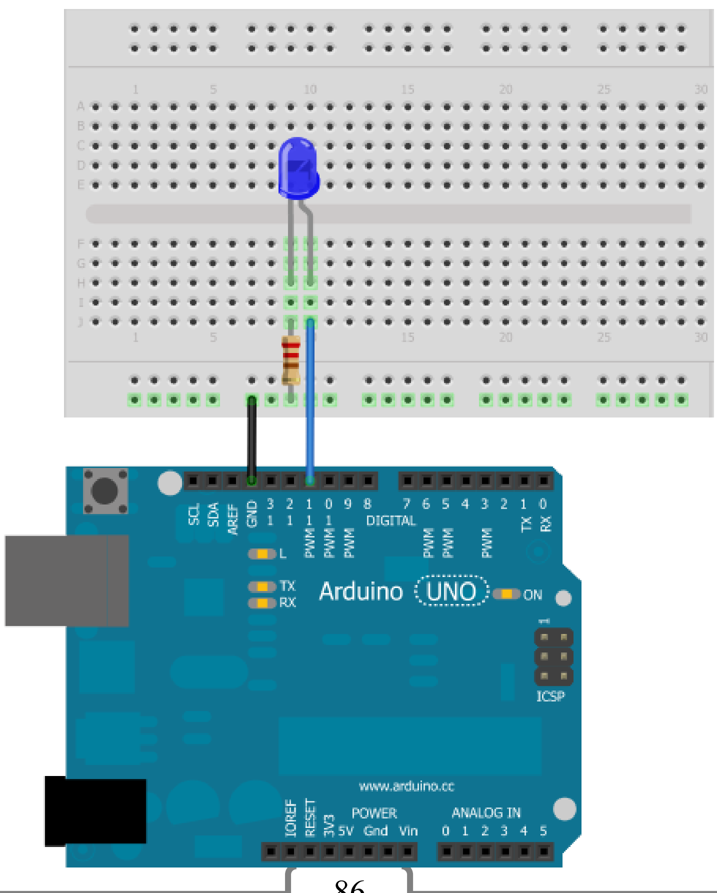
الكثير من المكونات الالكترونيه و الكهربيه تتفاعل مع القيم المختلفه من فرق الجهد بصور مختلفه مثل الدايود الضوئى، مثلا عندما تطبق 3 فولت على الدايود الضوئى ستجد ان اضائته ضعيفه و اذا رفعت الجهد الى 4 فولت ستجد الاضائه قد ازدادت و اذا و صلت الي 5 فولت ستجد الاضائه وصلت الى اعلى معدل سطوع.. و هناك مكونات كهربيه مثل

المحرك الكهربى (الموتور) و الذى تزداد سرعته بزياده الفولت الكهربى.

**في هـذا المثـال** وصلنا الدايود الضوئى على المخرج رقم 9 (الذى يدعم PWM) و استخدمنا مقاومه ضوئيه حتى تستشعر كميه الضوء الموجود حولها ثم ترسل الاشاره التماثليه الى اردوينو فيقوم بتشغيل الليد بناءا على الاشاره التماثليه من المقاومه الضوئيه .

لنأخذ مثال أخر على خاصيه التعديل الرقمي

## المثال السادس:المؤثرات الضوئيه بأستخدام PWM



في هذا المثال سنقوم بتوصيل دايود ضوئي ازرق (اخترت اللون الأزرق لان أضاءته عالية  مقارنه بالألوان الأخرى و يمكنك اختيار اللون الأبيض كذلك) إلى المنفذ رقم 11 و الذى يدعم خاصيه PWM و سنستغل تلك الخاصية في عمل بعض المؤثرات الضوئية الجميلة

و الأن نأتي إلى الكود البرمجي:

```
// Example_6_fading_led_light
const int ledPin = 11;
int i = 0;

void setup( )
  {
 pinMode(ledPin, OUTPUT);
   }
void loop()
{
    for (i = 0; i < 255; i++)
         {
    analogWrite(ledPin, i);
    delay(10);
          }
    for (i = 255; i > 0; i--)
        {
   analogWrite(ledPin, i);
   delay(10);
         }
}
```

العداد الأول

العداد الثاني

بعد الانتهاء من كتابه الكود و رفعه إلى المتحكمة الدقيقة.. لاحظ ماذا يحدث للدايود الضوئي و كيف تزداد إضاءته بهدوء و تنخفض بهدوء ☺

## سنتعلم الأن احدى اهم الأوام البرمجية و هي حلقات التكرار

في هذا المثال شاهدنا امر جديد و هو `for (i =0;  i < 10;  i++)` يستخدم هذا الأمر في عمل ما يسمى بالحلقة المتكررة loop ويمكن استخدام الحلقات فى أداء وظيفه تتكرر اكثر من مره بعدد معروف من المرات أو الي ما لا نهاية و يمكن استغلال الحلقات في عمل ما يسمى بالعدادات counters و هي برامج صغيره تقوم بالعد التزايدى أو التناقصي كما يريد المبرمج ، و تكون صيغه الأمر كتالي:

ماذا يفعل فى كل مره

هل يزداد العداد ام يقل

متى يتوقف

قيمه البداية

```
for (i=0; i<255; i++) {what to do here}
```

في المثال السابق استخدمنا عداد يقوم بعد الأرقام ابتدائياً من صفر حتى 255 و في كل مره يضيف واحد و يخزن القيمة في المتغير ( i ) ثم يقوم بتطبيق جهد تماثلي بقيمه i على الدايود الضوئي باستخدام الأمر `analogWrite(ledPin,i)` و بذلك نحصل على ذلك التأثير الضوئي الجميل حيث نرى الدايود الضوئي يبدأ في الوميض بصوره خافته ثم تزداد شده الإضاءة تدريجيا بزيادة قيمه المتغير ( i ) في العداد الأول.

**في العداد الثاني** استخدمنا نفس المبدأ لكن بصوره معكوسة بحيث يبدأ العداد من 255 و يظل الرقم يتناقص بمقدار واحد حتى يصل إلى الصفر و في كل مره يسجل العداد قيمه المتغير ( **i** ) و يطبق جهد تماثلي على الدايود الضوئي بقيمه( **i** ) و هذاما يجعل الدايود الضوئي يبدأ فى الوميض بشده ثم تنخفض شده إضاءته تدريجيا بتغير قيمه ( **i** )

**قد يتسائل البعض لماذا استخدمنا الأمر** delay(10); **داخل كل عداد ؟؟**

الأجابه تتمثل فى التحكم فى سرعه اردوينو اثناء تطبيق الجهد التماثلى و لكى يتضح المعنى بصوره عمليه جرب ان تغير الرقم و تجعله مره = 20 بحيث يكتب الأمر delay(20); و مره أخرى تجعله 50 و مره أخرى تجعله = 5 و شاهد ماذا سيحدث للدايود الضوئى.

## ملاحظات شخصيه :

هذه الصفحه مخصصه لكتابه ملاحظاتك الشخصيه عن الفصل الرابع :



# من أقوال العظماء:

العلــم مغرس كل فخــر فافتخــر

واحذر يفوتك فخر ذاك المغرس

واعلـم بأن العــلم ليس ينالــه

من همـه في مطعـم أو ملبس

الإمام الشافعى

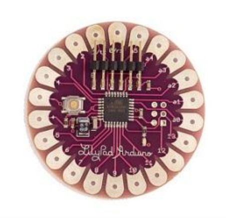
# الفصل الخامس

## أرسال البيانات و الأوامر من اردوينو إلى الحاسب الآلي و العكس

## Computer interfacing with Arduino



**جميع** بوردات اردوينو (بخلاف lilypad, mini) توفر إمكانيه أرسال و استقبال البيانات مباشره مع الحاسب الآلي computer interface و ذلك عن طريق منفذ الـUSB و يمكن استغلال هذه الخاصية في الكثير من المشاريع الرائع، وبالنسبة إلى البوردات التي لا تدعم هذه الخاصية فهي تحتاج إلى ما يسمى بالـ FTDI interface و هي شريحه صغيره تستخدم لتبادل البيانات بين الحاسب الآلي و المتحكمات الدقيقة



Liliypad

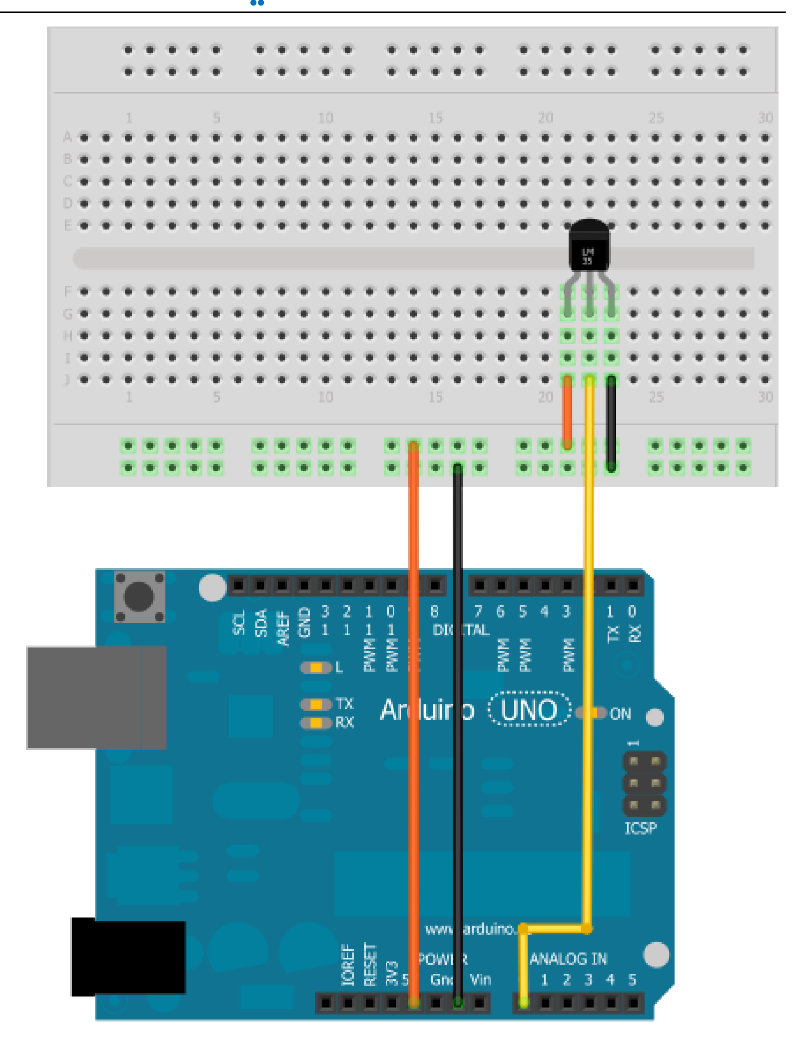


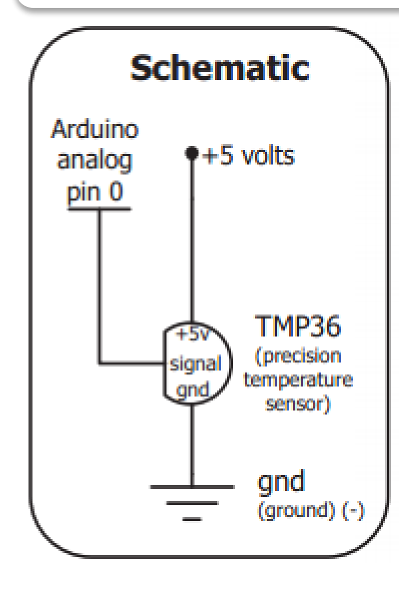
FTDI chip

**يتحدث** الفصل الخامس عن إمكانيه استخدام اردوينو في الحصول على معلومات من الحساسات المختلفة و أرسال النتائج لعرضها على الحاسب الآلي أو اخذ أوامر من الحاسب الآلي لتنفيذها على المكونات الإلكترونية الموصلة على البورده.

**في الأمثلة السابقة** كنا قد استخدمنا اردوينو في قراءة بعض الحساسات مثل الضوء و الحرارة و إظهار ناتج الحساس على دايود ضوئي ليوضح مقدار هذه القراءة أما الأن فسنستخدم خاصيه serial interface لأرسال نتائج القراءة الحساسات إلى الحاسب الآلي حيث يمكن قرائه النواتج بالأرقام و إجراء بعض العمليات الحسابية عليها.

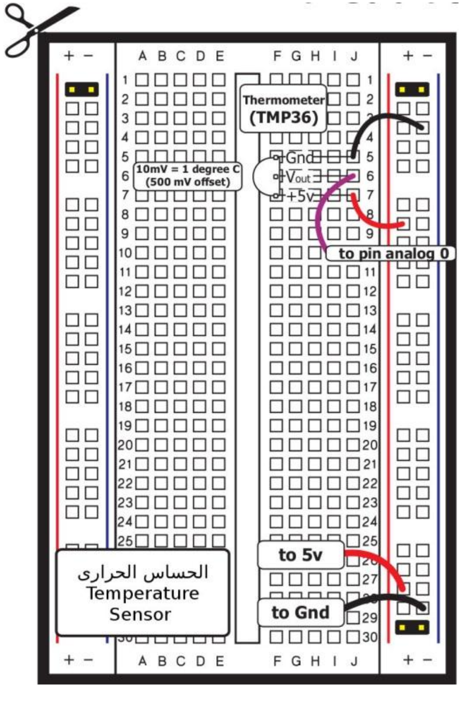
# المثال السابع: استخدام حساس الحرارة و أرسال درجه الحرارة إلى الحاسب الآلي



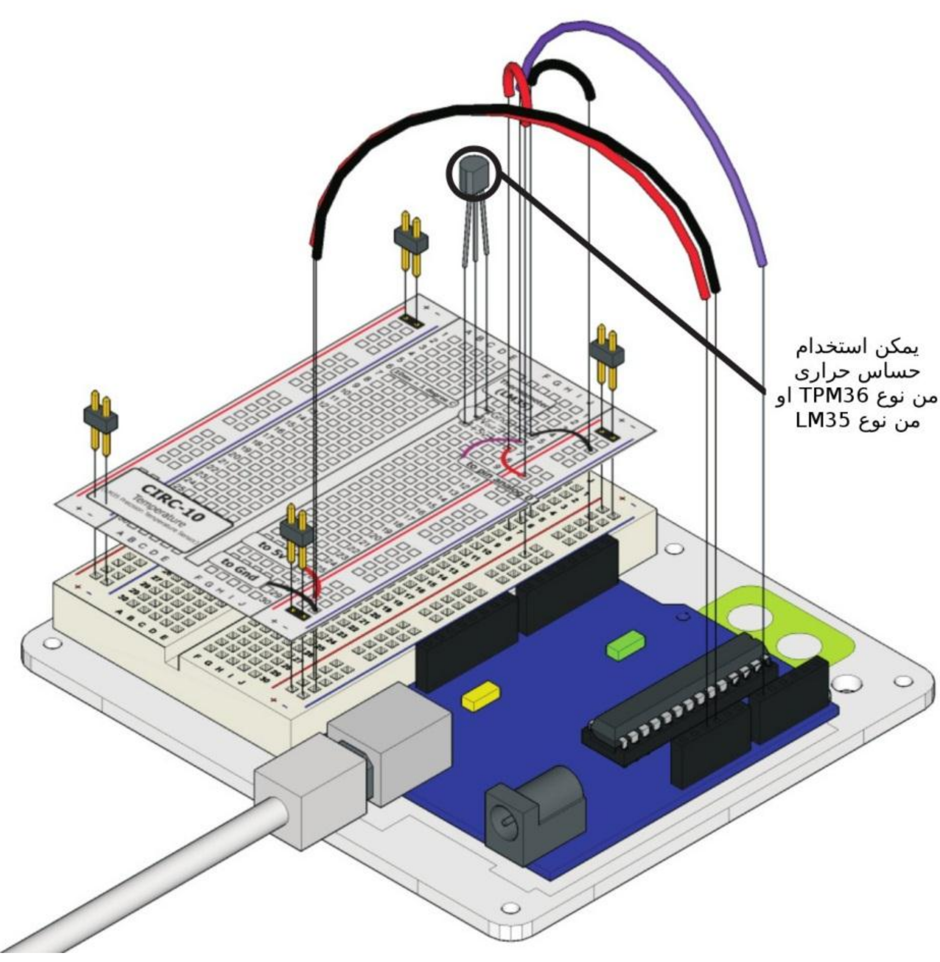



## مكونات المثال (7):

- ✓ بورده اردوينو Arduino Uno
- ✓ لوحه تجارب Breadboard
- ✓ حساس حراري TMP 36 وLM35
- ✓ أسلاك توصيل
- ✓ كابل التوصيل بالUSB
- ✓ ورقه المساعدة المطبوعة




لاحظ انه في حاله استخدامك لحساس حراري مختلف عن النوعين المذكورين في الأعلى فأن المداخل و المخارج الخاصة بالجهد قد تختلف و يمكنك معرفتها من خلال قراءة الداتا شيت الخاصة بالحساس المستخدم




**بعد الانتهاء من توصيل المكونات على لوحيه التجارب قم بكتابه الأكواد التالية و تأكد من صحتها عن طريق زر Verify ثم ارفعها إلى بورده اردوينو**

```
// Example_7_Temprature_Sensor
const int sensorPin = A0;
int reading;
float voltage;
float temperatureC;
void setup(  )
{ Serial.begin(9600); }
void loop (  )
 {
reading = analogRead(sensorPin);
voltage = reading * 5.0/1024;
Serial.print(voltage);
Serial.println(" volts");
temperatureC = (voltage - 0.5) * 100 ;
Serial.println("Temperature is: ");
Serial.print(temperatureC);
 Serial.println(" degrees C");
delay(1000);
 }
```

بعد الانتهاء من كتابه و رفع الكود البرمجي إلى المتحكمة نقوم بالضغط على زر Serial Monitorالموجـــود في شـــريط الأوامـــر الســـريعة كمـــا في الصـــورة التاليـــة:

sketch_jun18a | Arduino 1.0.1

File Edit Sketch Tools Help

Serial Monitor

sketch_jun18a §

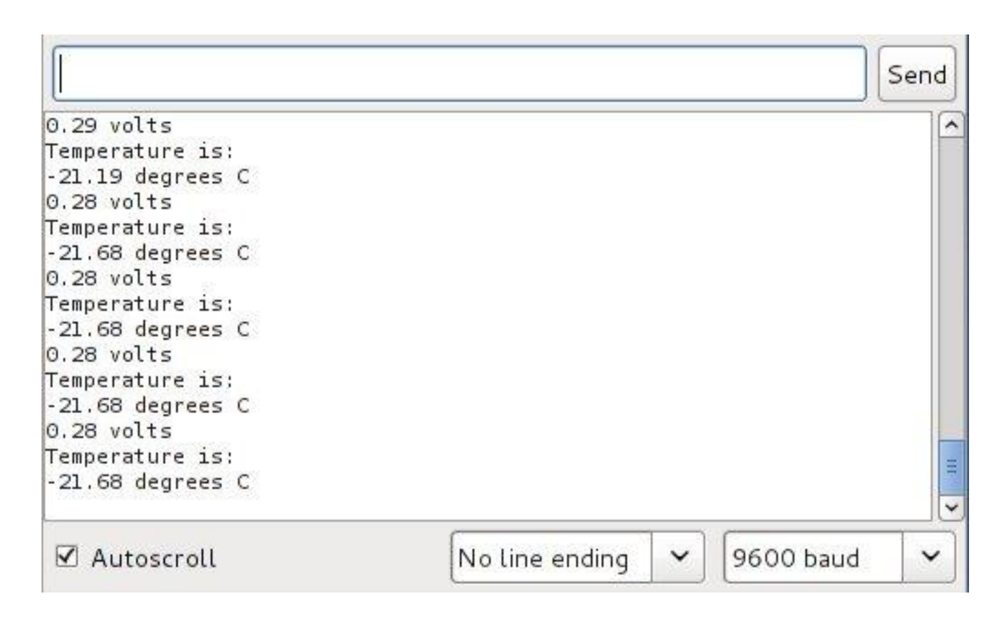
```
int sensorPin = A0;
int reading;
float voltage;
float temperatureC;

void setup(  )
{ Serial.begin(9600); }

 void loop (  )
 {
reading = analogRead(sensorPin);
voltage = reading * 5.0;
voltage /= 1024.0;
Serial.print(voltage);
Serial.println(" volts");
temperatureC = (voltage - 0.5) * 100 ;
Serial.print(temperatureC); Serial.println(" degrees C");
delay(1000);
 }
```

20 Arduino Uno on COM1

ستظهر صفحه خاصه تستقبل البيانات من بورده اردوينو و تقوم بعرض درجه الحرارة كل ثانيه كما سنلاحظ أن الدايود الضوئي الموجود على بورده اردوينو المكتوب بجانبه Tx بدأ في الوميض بسرعه و ذلك يعنى أن اردوينو يرسل بيانات إلى الحاسب الآلي ...




الأن حاول ن ترفع درجه حراره الحساس عن طريق تقريب شمعه مشتعلة أو عود ثقاب مشتعل أو أي مصدر حرارى متوفر (حذارى أن تمس النار الحساس الحرارى فذلك قد يتلفه ، هناك حساسات حرارية خاصه يمكنها تحمل الحرارة الشديدة لكن الحساس المستخدم في هذا المثال يستطيع تحمل حراره تصل إلى 150 درجه مئوية تقريبا )

ملحوظه: علامه الشرطة ( - ) التي تكتب قبل درجه الحرارة لا تعنى سالب درجه مئوية و إنما خطأ برمجي مؤقت في أول السطر عند استقبال بيانات من اردوينو

# شرح الكود البرمجي

**يستخدم الأمر** `Serial.begin(9600);` في جعل بورده اردوينو تبدأ التواصل مع الحاسب الآلي عن طريق منفذ الـUSB و بهذا يمكن للاردوينو أرسال أو استقبال بيانات من و إلى الحاسب .

نلاحظ انه هناك متغيران و هما voltage و temperatureC تم تعريفهم باستخدام الأمر float بدلا من استخدام الأمر int و ذلك لان الحساس الحرارى يقيس درجه الحرارة بدقه عالية تصل إلى 0.1درجه مئوية و من المتوقع أن تكون قيمه الجهد الكهربي الناتج عنه و درجه الحرار بالكسور العشرية و ليس الأعداد الصحيحة لذلك استخدامنا الأمر float لجعل هذه المتغيرات تقبل قيمه تحتوى على كسور عشريه .

يستخدم الأمر `reading = analogRead(sensorPin);` فى تسجيل قراءة الدخل التماثلي على المتغير reading و هي القيمة الناتج من sesnorPin و الذى يرمز إلى المدخل A0 .

كما ذكرنا سابقا ان المتحكمة الدقيقة تقوم بتحويل الإشارة التماثلية إلى قيمه رقميه من صفر الى 1024 لذلك استخدمنا الأمر التالي لتحويل هذه القيمة الرقمية إلى قيمه فولتيه نستطيع فهمها

```
voltage = reading * 5/1024;
```

**بعد تحويل** القيمة الرقمية إلى قيمه فولتيه استخدمنا الأمر `Serial.print(voltage);` لأرسال هذه القيمة إلى الحاسب الألي لعرضها على بيئة التطوير باستخدام الـ Serial Monitor ثم نستخدم الأمر `Serial.println ("voltage");` ليقوم بطابعه كلمه voltage بعد كل قيمه فولت يعرضها على شاشه الحاسب الآلي

نستخدم الأمر `temperatureC = (voltage - 0.5) * 100;` لتحويل قيمه الفولت إلى درجه حراره بوحده سي لزيوس Celsius ثم نكرر نفس أوامر Serial.print مع درجه الحرارة لعرضها على شاشه الحاسب الآلي

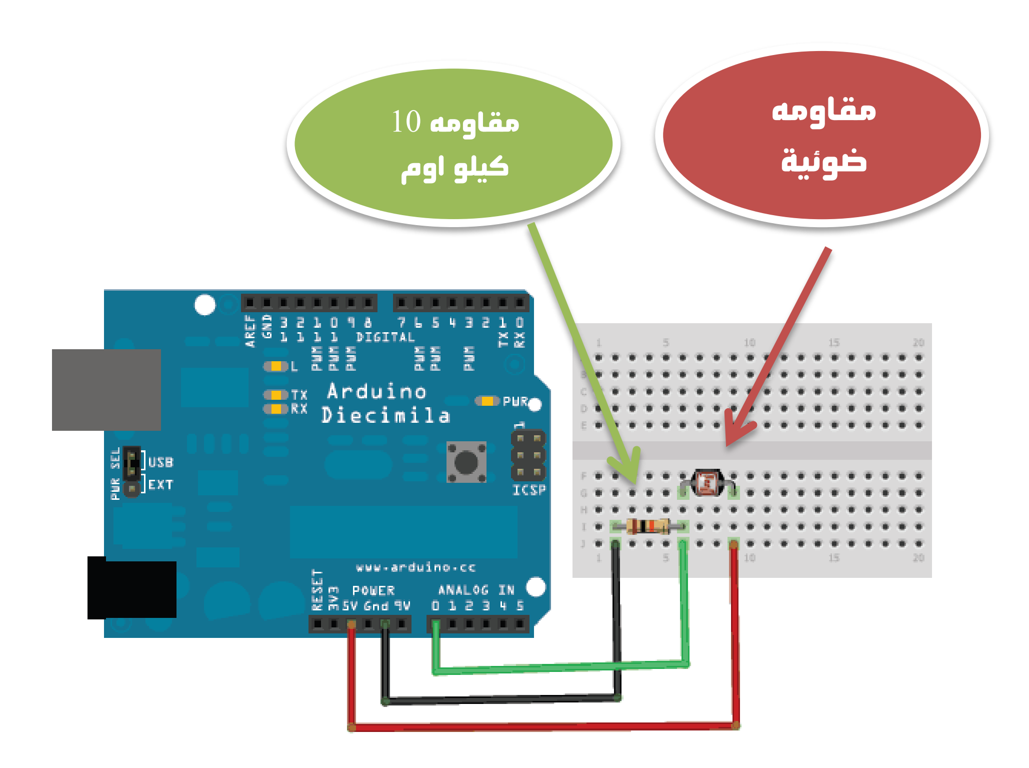
```
Serial.print(temperatureC);

 Serial.println(" degrees C ");
```

الأمر الأخير هو `delay(1000);` و ذلك لنجعل المتحكمة الدقيقة تنتظر ثانيه واحده قبل أن تعيد أرسال درجه الحرارة و الفولت إلى الحاسب الآلي مره أخرى.

# المثال الثامن : استخدام حساس الضوء و عرض شده الإضاءة على الحاسب الآلي

هل تتذكر المثال الخامس، سنقوم بأعاده تنفذ ذلك المثال ولكن بدلا من عرض شده الإضاءة على الدايود الضوئي led ، سنعرضها على الحاسب الآلي .



```
// Example_8_Light_Sensor_2

const int photocellPin = A0;

int photocellReading;

void setup(void)

{ Serial.begin(9600); }



void loop(void)

 {

   photocellReading = analogRead(photocellPin);

   Serial.print("Analog reading = ");

   Serial.print(photocellReading);



if (photocellReading < 10) { Serial.println(" - Dark");}

else if (photocellReading < 200) { Serial.println(" - Dim");}

else if (photocellReading < 500) {Serial.println(" - Light"); }

else if (photocellReading < 800) { Serial.println(" - Bright"); }

else {Serial.println(" - Very bright"); }

  delay(1000);

}
```

**بعـد الانتهـاء** مـن كتابـه و رفـع الكـود البرمجـي إلـى المتحكمـة نقـوم بالضغـط علـى زر Serial Monitor الموجود في شريط الأوامر السريعة كما في الصورة التالية :

sketch_jun18a | Arduino 1.0.1

File Edit Sketch Tools Help

Serial Monitor

sketch_jun18a §

```
int photocellPin = A0;
int photocellReading;
void setup(void)
{ Serial.begin(9600); }

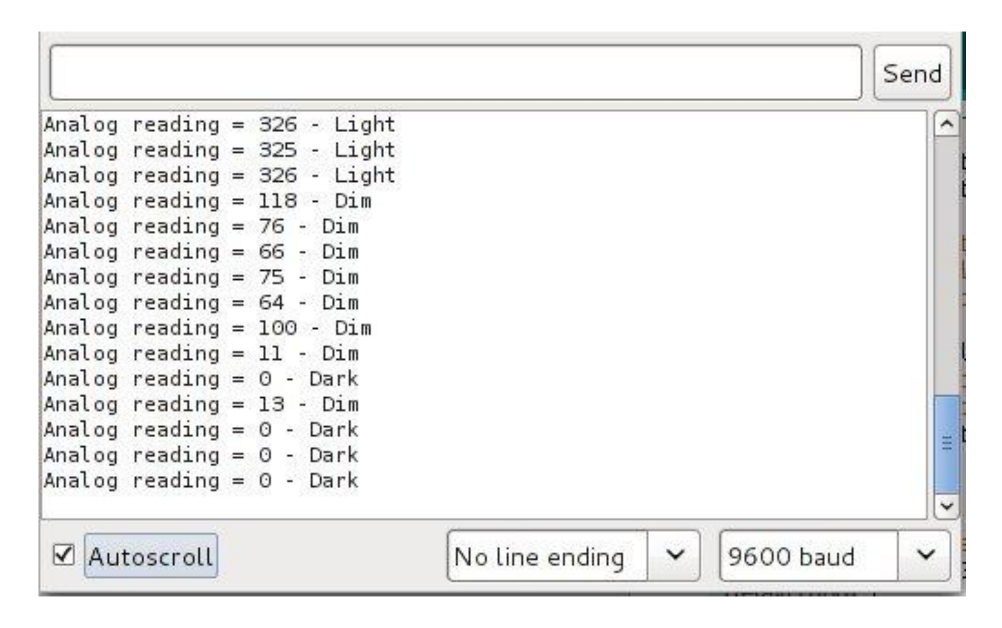
void loop(void)
 {
   photocellReading = analogRead(photocellPin);
   Serial.print("Analog reading = ");
   Serial.print(photocellReading);

if (photocellReading < 10) { Serial.println(" - Dark");;}
else if (photocellReading < 200) { Serial.println(" - Dim");;}
else if (photocellReading < 500) {Serial.println(" - Light"); }
else if (photocellReading < 800) { Serial.println(" - Bright"); }
else {Serial.println(" - Very bright"); }
  delay(1000);
}
```

19 Arduino Uno on COM1

**الأن حاول** أن تغير مستوى الإضاءة الذى تتعرض له المقاومة الضوئية عن طريق :

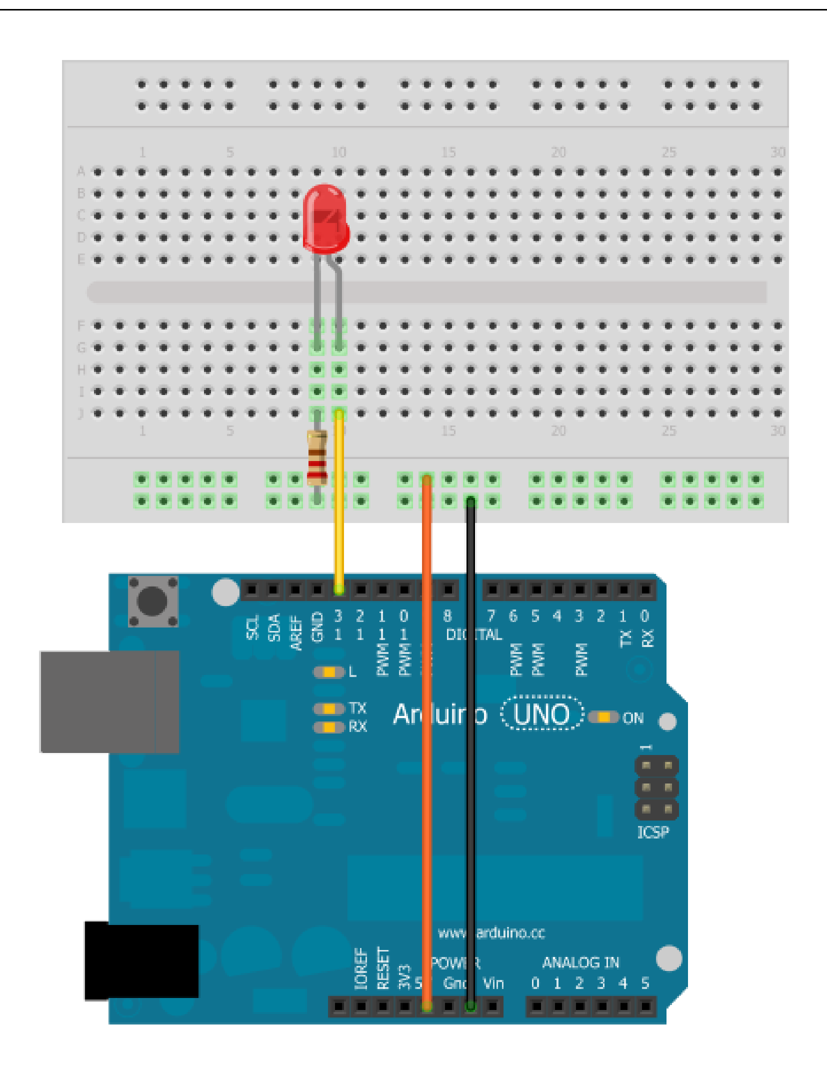
- تعريضها للضوء الساطع مباشره
- تغطيتها بقطعه قماش خفيفة
- تغطيتها بيديك لتحجب عنها الضوء تماما



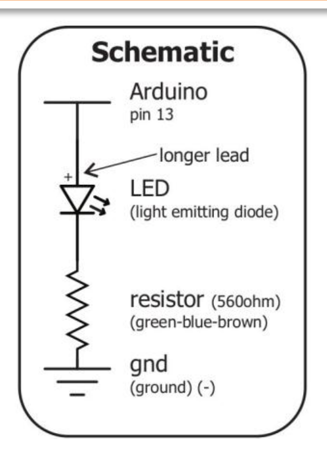

**ملحوظه:**

- Dim تعنى إضاءة خافته
- Dark تعنى ظلام دامس
- Light تعنى إضاءة معتدلة
- Bright Light تعنى إضاءة شديده

# المثال التاسع: تشغيل دايود ضوئي عن طريق استقبال أمر من الحاسب الآلي



## مكونات المثال (9):



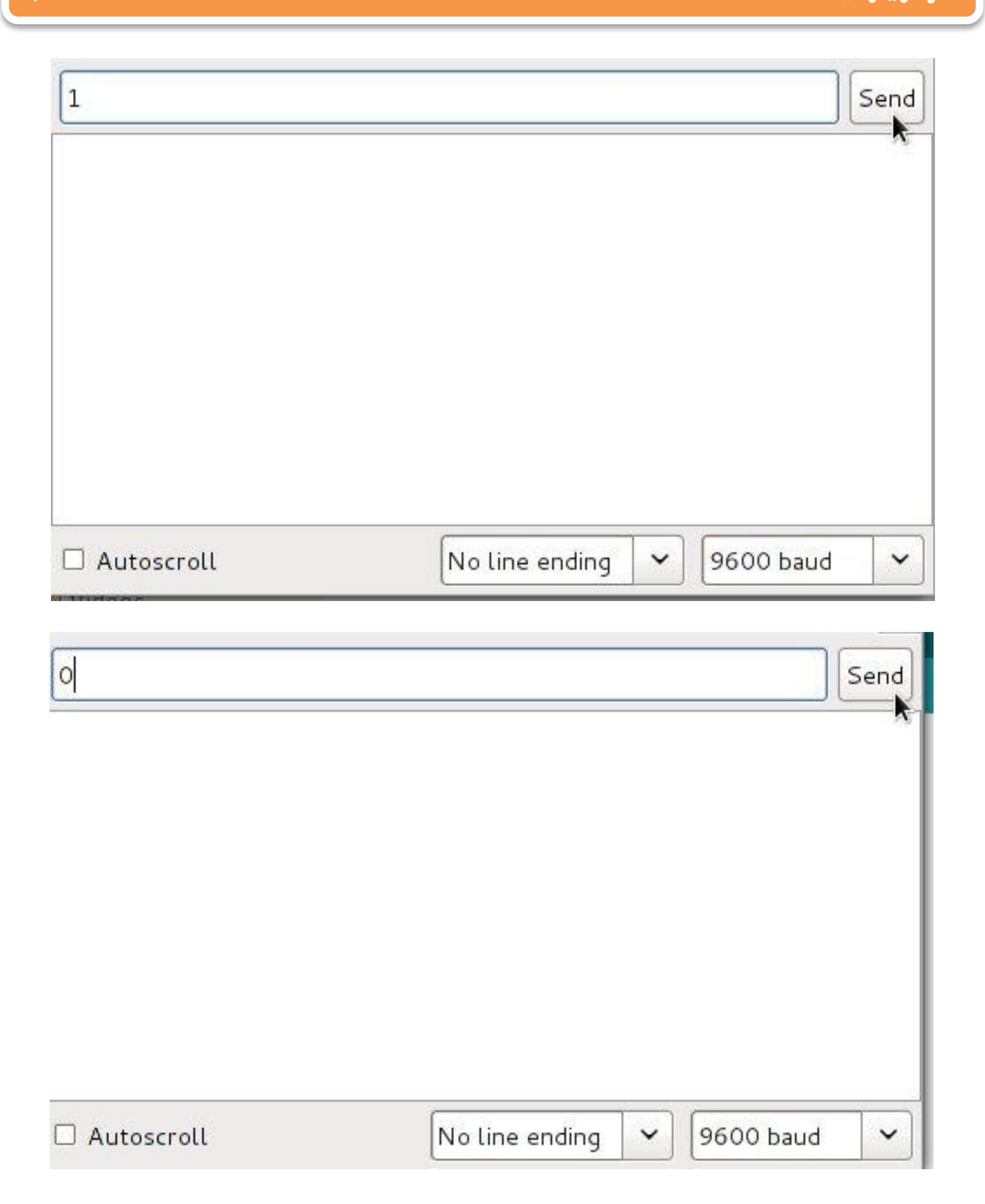

- ✓ بورده اردوينو Arduino Uno
- ✓ لوحه تجارب
- ✓ دايود ضوئي led 5mm
- ✓ مقاومه 560 اوم
- ✓ أسلاك توصيل

**يعتبر المثال** التاسع تطوير للمثال الأول و الثاني ، و في هذا المثال سنستخدم الحاسب الآلي في التحكم في الدايود الضوئي بدلا من السويتش، سنجعل اردوينو يستقبل امر التشغيل و الإطفاء عن طريق الـUSB باستخدام خاصيه الـ Serial Monitorفي بيئة تطوير اردوينو Arduino IDE

الكود:

```
// Example_9_Computer_Interfacing
int ledPin=13;
int value;
void setup ()
{
  Serial.begin(9600);
  pinMode(ledPin,OUTPUT);
}
void loop ()
{
value = Serial.read();
if (value == '1') {digitalWrite(ledPin,HIGH);}
else if (value == '0') {digitalWrite(ledPin,LOW);}
}
```

**بعد الانتهاء** من كتابة الكود البرمجي و رفعه إلى اردوينو اضغط على أيقونة serial Monitor ستجد شريط كتابي في اعلى الصفحة قم بكتابه 1 و اضغط زر أرسال Send و شاهد ماذا سيحدث للدايود الضوئي ، ثم اكتب الرقم صفر و اضغط على زر أرسال مره أخرى و شاهد ماذا سيحدث ...


1
Send
Autoscroll
No line ending
9600 baud
0
Send
Autoscroll
No line ending
9600 baud

**في هـذا** المثال استخدمنا الأمر ;( )Serial.read و هـو الامـر المستخدم فـى قـراءة البيانات المرسلة من الحاسب الآلي إلى اردوينو عبر منفذ الـUSB و قمنا بإضافة =value قبل هذا الأمر و ذلك حتى تقوم المتحكمة بقراءة ما يرسل من الـ USB، تخزن هذه القيمة في المتغير value

**ثم استخدمنا** جمله if.. else if لوضع شرط:

- اذا كانت قيمه المتغير value == 1 تقوم المتحكمة بتشغيل الدايود الضوئي
- أما اذا كانت قيمه المتغير value == 0 تقوم المتحكمة بإطفاء الدايود الضوئي

## الأن حاول أن تعدل الكود بنفسك و تزيد عدد المخارج التي يمكن التحكم بها عن طريق الحاسب الآلي

### مصادر اضافيه للمعلومات:

لمزيد من المعلومات عن الأوامر المستخدمة في الـ Serial communication تفضل الرابط التالي لمرجع الأوامر الرسمي:

http://arduino.cc/en/Reference/serial

## ملاحظات شخصيه :

هذه الصفحه مخصصه لكتابه ملاحظاتك الشخصيه عن الفصل الخامس :



# من أقوال العظماء:

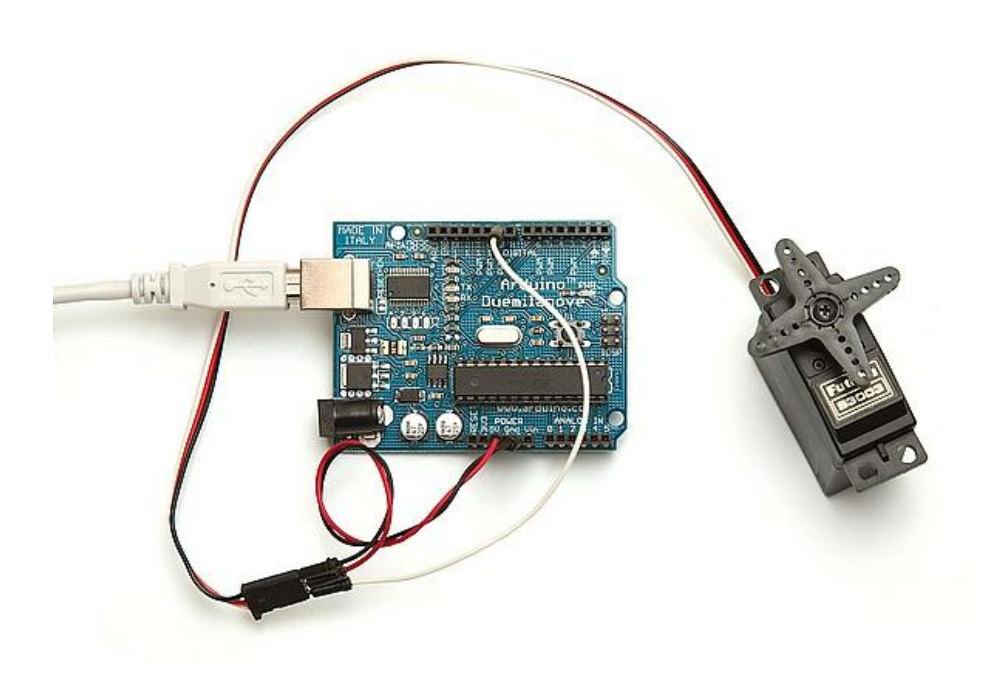
**المعرفه قوه فى يد من يعلمها**

ألبرت أينشتين

# الفصل السادس

## أستخدام المحركات

## The Motors





تعتبر المحركات من أهم العناصر الداخله فى المشاريع الألكترونيه و هى العنصر المسؤل عن تحويل الطاقه الكهربيه الى طاقه حركيه فى صوره دوران .

ستجد المحركات فى كثير من الأجهزه الالكترونيه التى يوجد بها حركه ميكانيكيه مثل: الروبتات بأنواعها المختلفه سواء كانت صناعيه او روبوتات للترفيه، مشغلات الأقراص، ألعاب الأطفال.. الخ



تنقسم المحركات الكهربيه الى نوعين رئيسيين و هما :

- محركات التيار المستمر (DC – Servo – Stepper)
- محركات التيار المتردد (3 Phase –1 Phase)

فى هذا الفصل سنتعرض للنوع الأول و هو المحركات التى تعمل بالتيار المستمر مثل الـ DC Motor و الـ Servo Motor و هما اشهر أنواع المحركات المستخدمه فى مشاريع المتحكمات الدقيقه بأنواعها المختلفه.

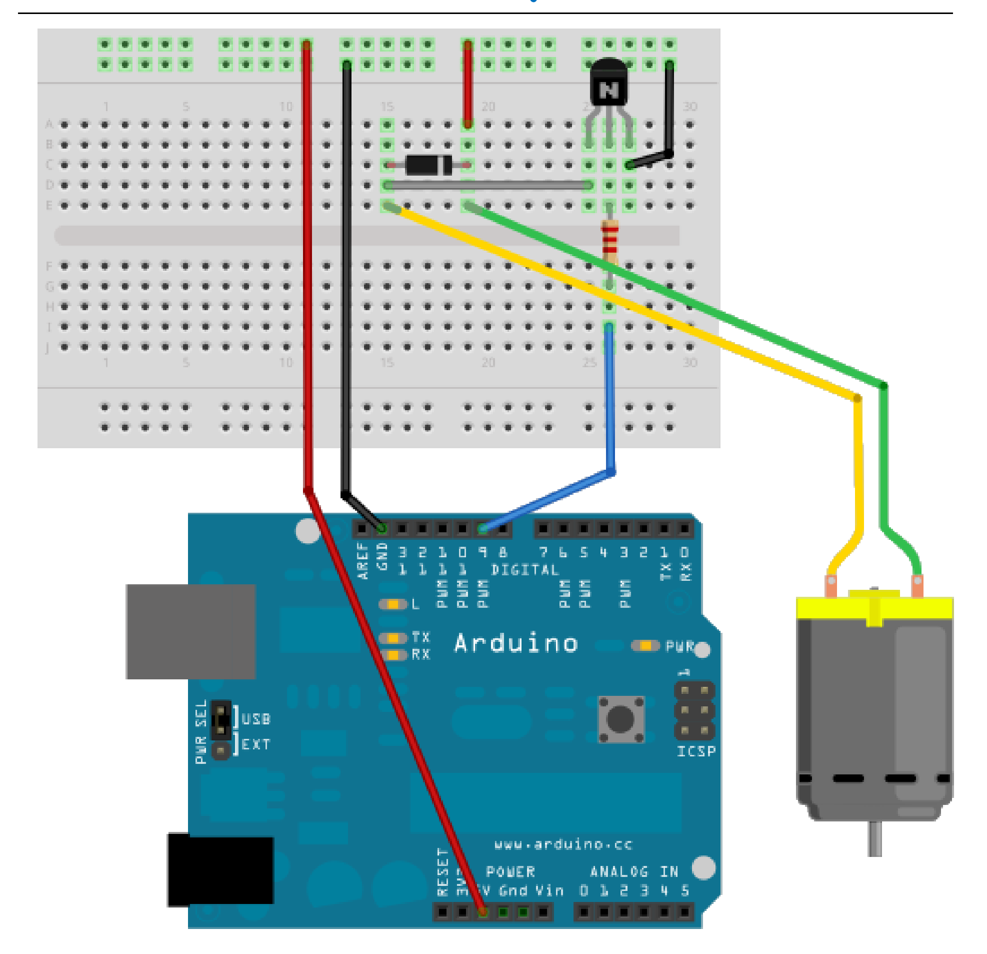


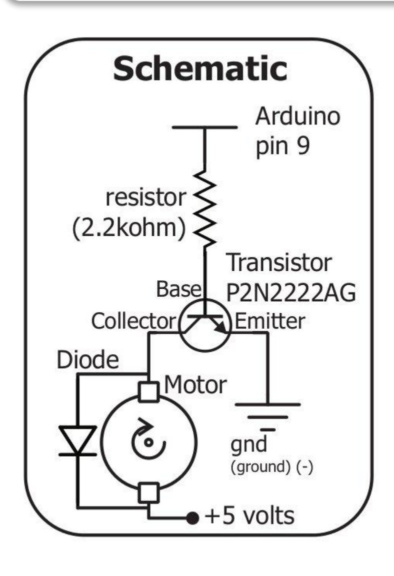



Servo Motor

DC Motor

# المثال العاشر: استخدام محرك التيار المستمر DC

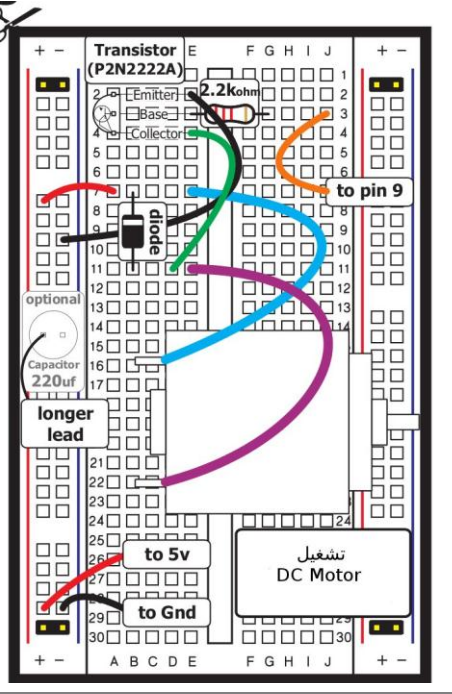




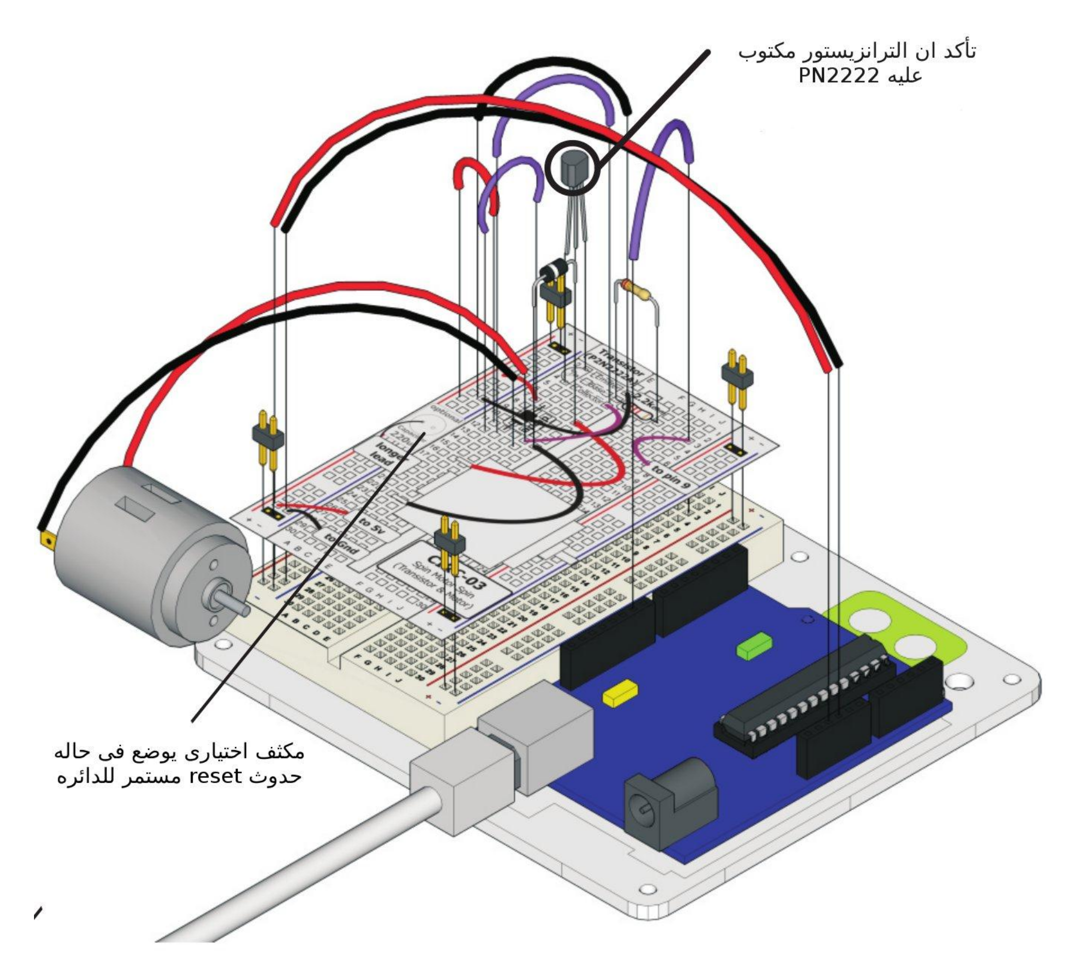

## مكونات المثال (7):

- ✓ بورده اردوينو Arduino Uno
- ✓ لوحه تجارب Breadboard
- ✓ محرك تيار مُستَمرصغير DC motor
- ✓ ترانزستور 2N2222 او PN2222
- ✓ دايود 1N4001 او اى بديل
- ✓ مقاومه 2.2 كيلو اوم
- ✓ أسلاك توصيل
- ✓ كابل التوصيل بالـUSB
- ✓ ورقه المساعدة المطبوعة

---




فى هذا المثال سوف نستخدم محرك تيار مستمر من الحجم الصغير و الذى يوجد عاده فى لعب الأطفال و يعمل بفرق جهد يبدأ من 3 فولت و اقصاه 9 فولت و ستجد مثل هذه المحركات متوفره فى محلات المكونات الألكترونيه او فى اى من لعب اطفال القديمه التى تحتوى على محركات ☺




**بعد الانتهاء من توصيل المكونات على لوح التجارب قم بكتابه الأكواد التالية ثم ارفعها إلى بورده اردوينو**

```
//Example_10_DC_Motor
int motorPin = 9 ;
int onTime = 2500 ;
int offTime = 1000 ;

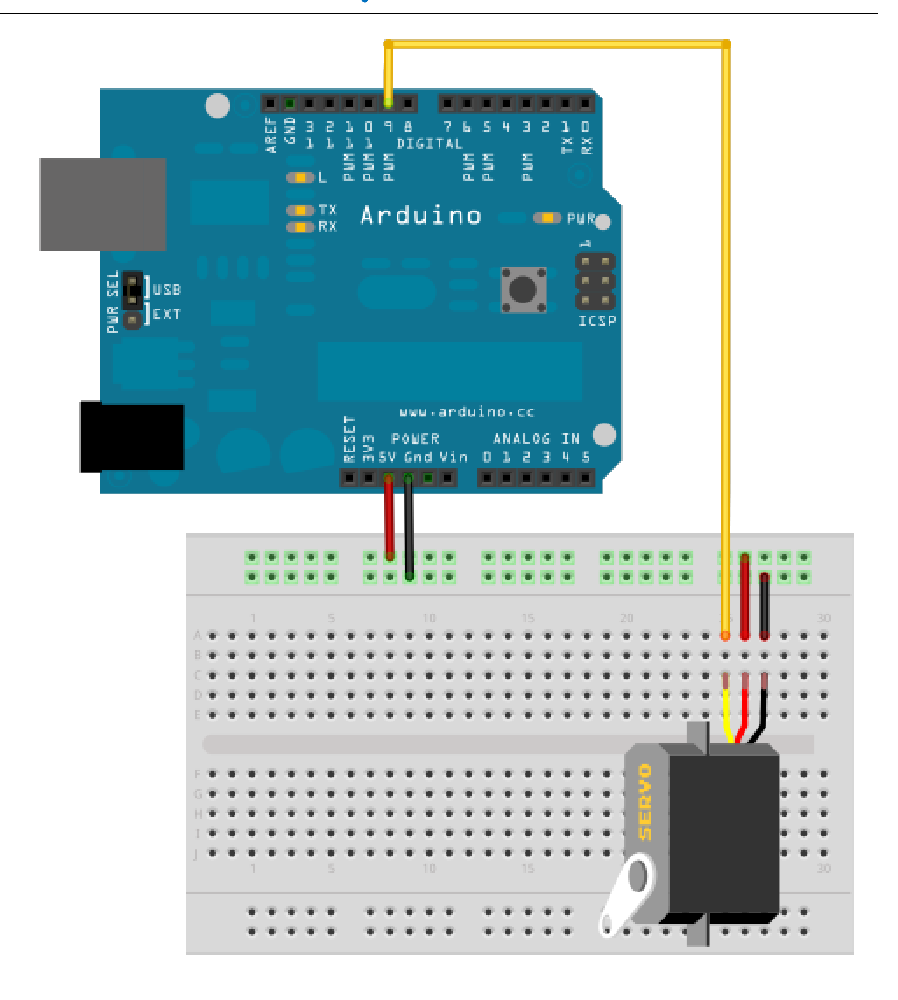
void setup ( )
{pinMode(motorPin, OUTPUT); }

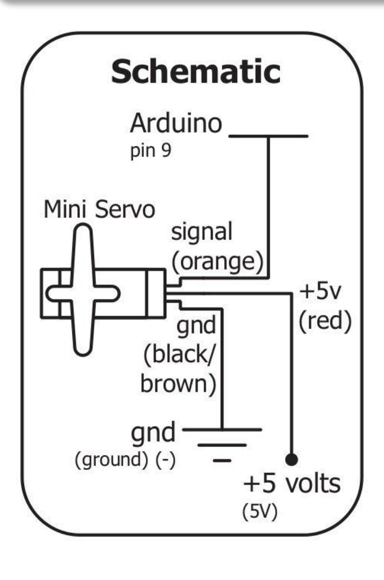
void loop ( )
{
analogWrite(motorPin,100);
delay(onTime);
digitalWrite(motorPin, LOW);
delay(offTime);

analogWrite(motorPin,190);
delay(onTime);
digitalWrite(motorPin, LOW);
delay(offTime);

analogWrite(motorPin,255);
delay(onTime);
digitalWrite(motorPin, LOW);
delay(offTime);
}
```

## المثال الحادى عشر: استخدام محرك سيرفو

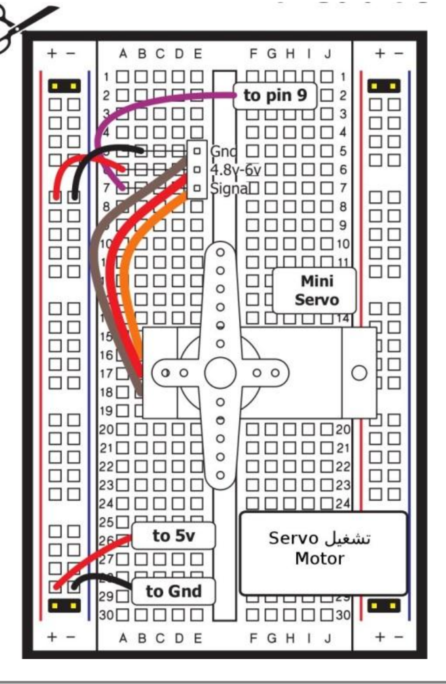




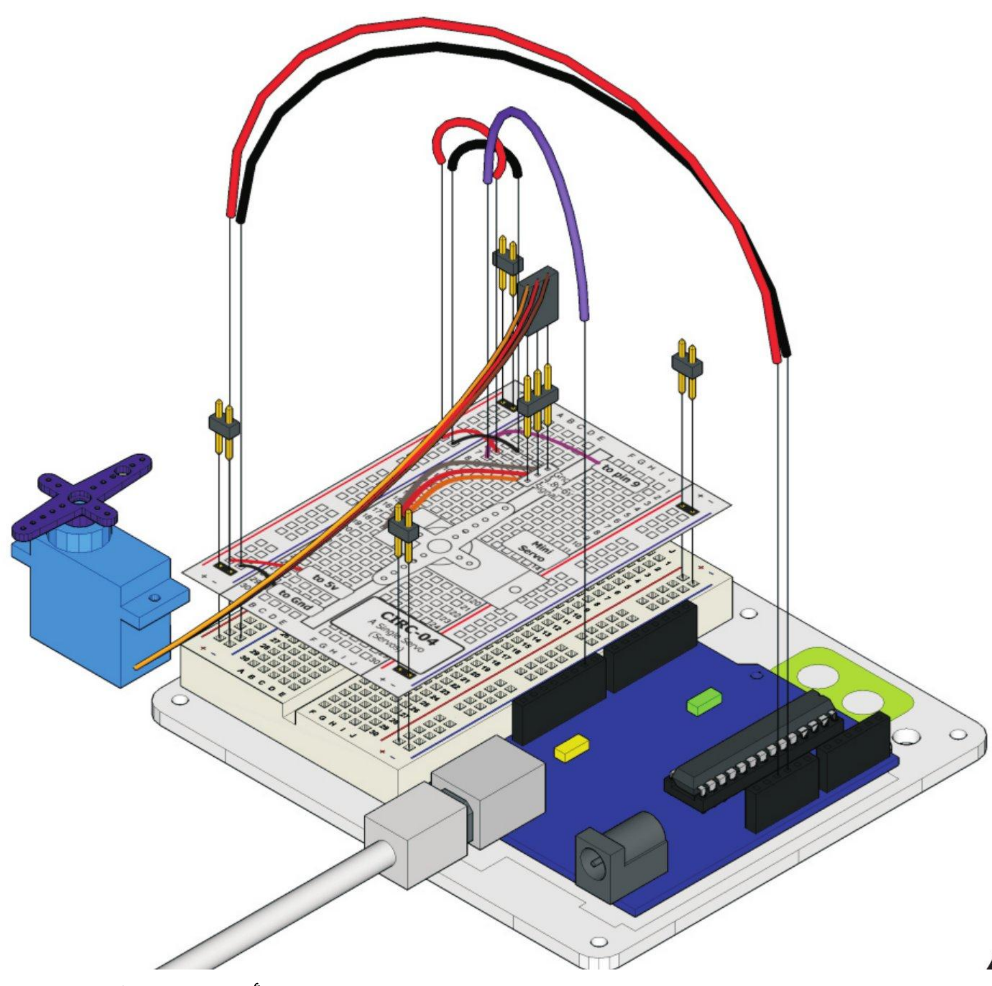

## مكونات المثال (7):

- ✓ بورده اردوينو Arduino Uno
- ✓ لوحه تجارب Breadboard
- ✓ محرك تيار مُستَمرصغير DC motor
- ✓ ترانزستور 2N2222 او PN2222
- ✓ دايود 1N4001 او اى بديل
- ✓ مقاومه 2.2 كيلو اوم
- ✓ أسلاك توصيل
- ✓ كابل التوصيل بالـUSB
- ✓ ورقه المساعدة المطبوعة

---




فى هذا المثال سوف نستخدم محرك تيار مستمر من الحجم الصغير و الذى يوجد عاده فى لعب الأطفال و يعمل بفرق جهد يبدأ من 3 فولت و اقصاه 9 فولت و ستجد مثل هذه المحركات متوفره فى محلات المكونات الألكترونيه او فى اى من لعب اطفال القديمه التى تحتوى على محركات ☺




بعد الانتهاء من توصيل المكونات على لوح التجارب قم بكتابه الأكواد التالية ثم ارفعها إلى بورده اردوينو (ملحوظه: يمكنك ان تجد الكود البرمجى فى قائمه الأمثله الجاهزه فى برنامج اردوينو  examples - servo - sweep)

```
//Example_11_Servo_Motor
#include <Servo.h>
Servo myservo;
int pos = 0;

void setup()
{
  myservo.attach(9);
}

  void loop()
{

  for(pos = 0; pos < 180; pos += 1)
  {
    myservo.write(pos);
    delay(15);
  }
  for(pos = 180; pos>=1; pos-=1)
  {
    myservo.write(pos);
    delay(15);
  }

}
```

مكتبه اضافيه

## ملاحظات شخصيه :

هذه الصفحه مخصصه لكتابه ملاحظاتك الشخصيه عن الفصل السادس :

اكتب ملاحظاتك
هنا

## من أقوال العظماء :

الجبال قد تصمُد امام الزلازل لكنها لن تقاوم
قَطرات الماء التى تَهطُل بأنتظام فى هدوء و
تكرار ، فأجعل عَملك مثل قطرات المياه حتى
تتغلب على جميع العوائق

حكمة صينية قديمة

# الفصل السابع

## وسائل الادخال و الاخراج المتطورة

## Advanced Inputs/Outputs





**فى الفصول** السابقه اعتمدنا بشكل اساسى على ابسط وسائل الادخال و الاخراج للتعامل مع اردوينو مثل السويتش( ادخال Input ) و الدايود الضوئى (اخراج output)

في هذا الفصل سنستعرض بعض المكونات الاكثر تطورا للتفاعل مع المتحكمات الدقيقه في مشاريع واقعيه يمكن استخدامها فى حياتنا اليوميه، سيتطرق هذا الفصل الى شرح المكونات التاليه:

- شاشات عرض الكريستال السائل liquid crystal Display: LCD
- لوحه ادخال الارقام Keypad
- مصفوفه الدايود الضوئى Led matrix
- المتممات Relays





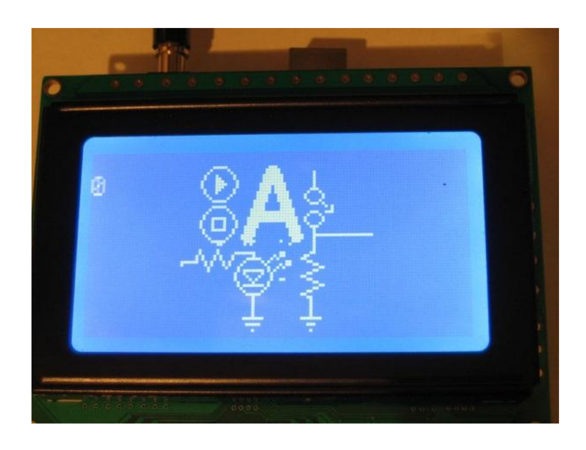



**شاشات عرض الكريستال السائل :** تتكون هذا الشاشات من زجاج الكريستال المعالج و تتوفر هذه الشاشات بأحجام و انواع مختلفه و سوف نستعرض منها نوعين اساسيين و هما:

- شاشات العرض المعتمده على الحروف Character LCD
- شاشات العرض المعتمده على الرسومات Graphical LCD

Character LCD

Graphical LCD



تـوفر شاشات عرض المعتمده على الحروف Character LCD امكانيه اخراج اى نصوص تتكون من حروف او ارقام او رموز (مثل التى تكتب على لوحه المفاتيح فى الحاسب الآلى ) و تتوفر بأحجام مختلفه و الوان مختلفه مثل:

Green 16x2 lcd
Blue 16x2 lcd
Grees 20x4 lcd

و يمثل الرقم 16x2 عدد السطور (2) الحروف التى يمكن كتابتها فى كل سطر (16) حرف كما تتوفر بألوان مختلفه كما فى الصور التاليه:



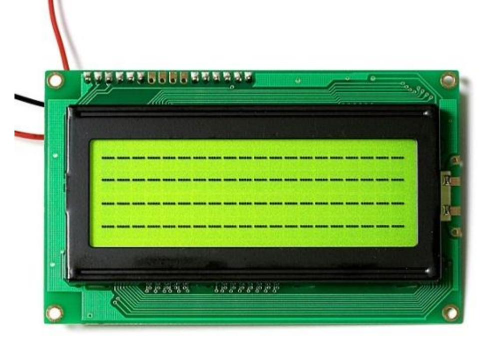

Blue 16x2 LCD




Green 16x2 LCD



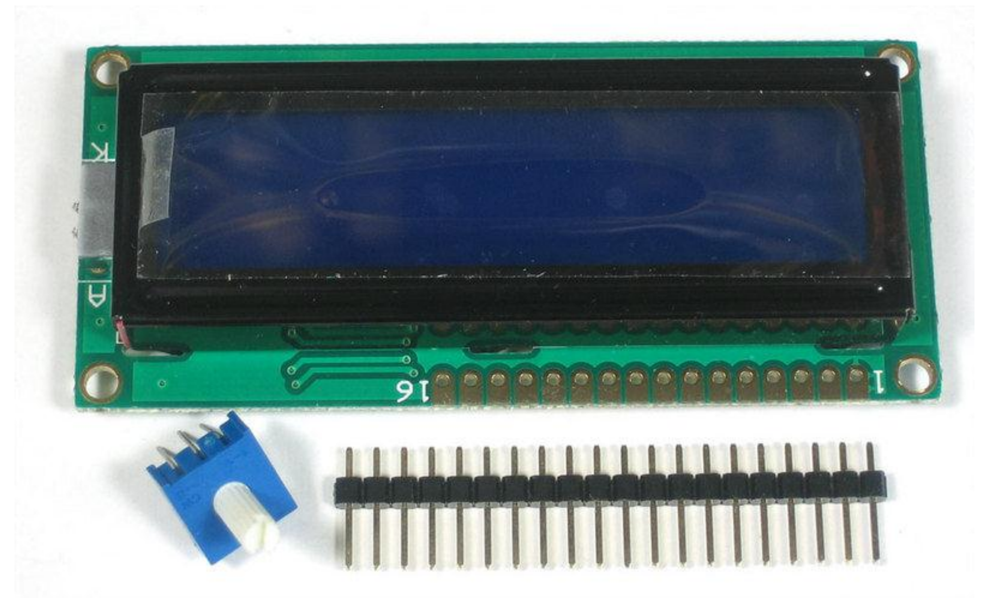

Black 16x2 LCD





Green 20x4 LCD

# المثال العاشر: توصيل شاشه Character LCD بمقاس 16x2

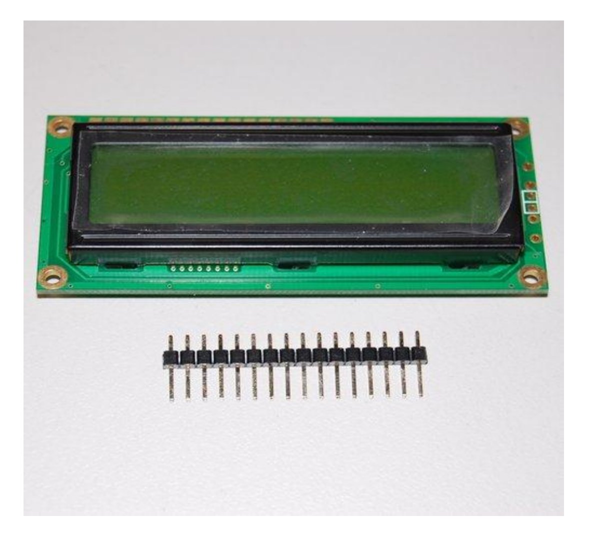


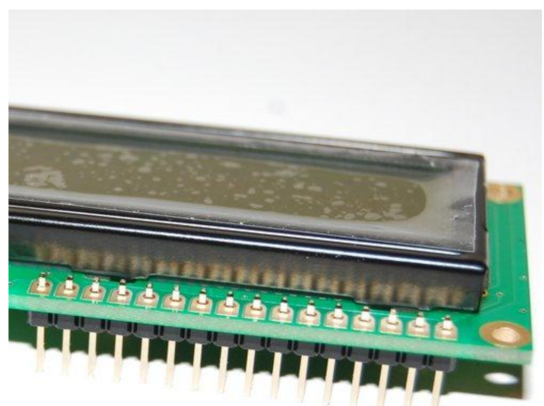

**سنحتاج المكونات التاليه:**

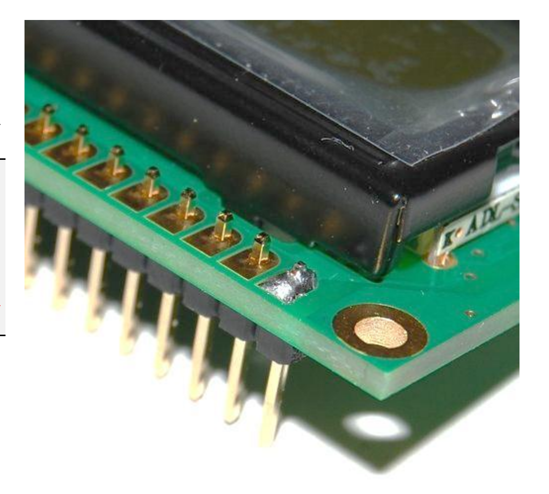
- شاشه بقياس 16x2 لون اضائه ازرق (او اى لون تفضله انت )
- اعمده توصيل نحاسيه قصيره 16 نقطه copper Pin Headers 16 point
- مقاومه متغيره بقيمه 10 كيلو اوم
- مكواه لحام
- قصدير لحام

الخطوة الأولى هي لحام نقاط التوصيل Pin Headers بنقاط لحام الشاشه و يتم استخدام مكواه اللحام و القصدير فى هذا العمليه:

| **اذا اردت تعلم لحام المكونات الإلكترونية توجهه إلى اللينك التالي**<br>http://www.aaroncake.net/electronics/solder.htm |
|---|







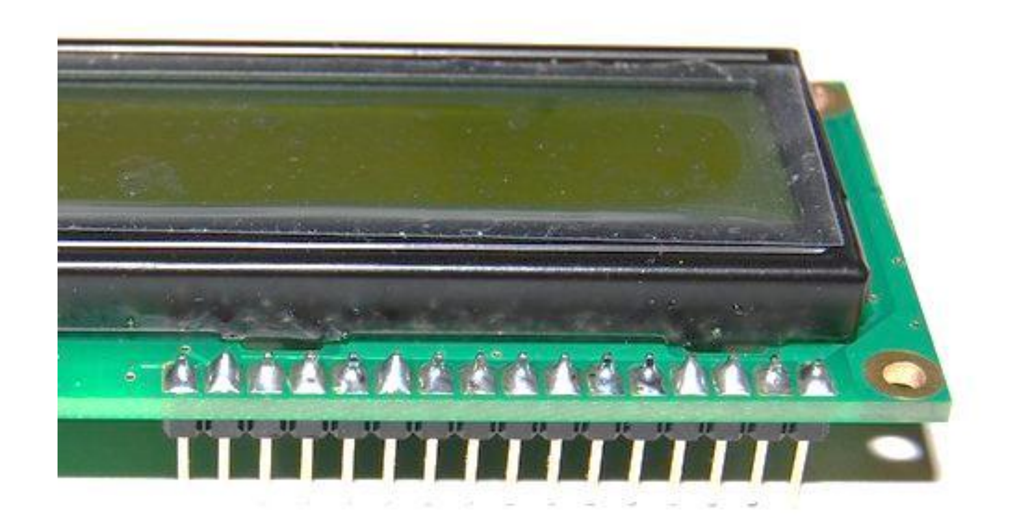
قـم بلحـام اول نقطـه توصـيل باسـتخدام الكاويه و القصـدير و انتظـر 5 ثـوانى حتـى تبرد نقطه التوصيل

تنبيـه: لا تحـاول لمـس نقطـه اللحـام بيـدك بعـد اللحـام مباشـره فقـد يـؤدى ذلـك الى حدوث حروق للجلد من حراره اللحام

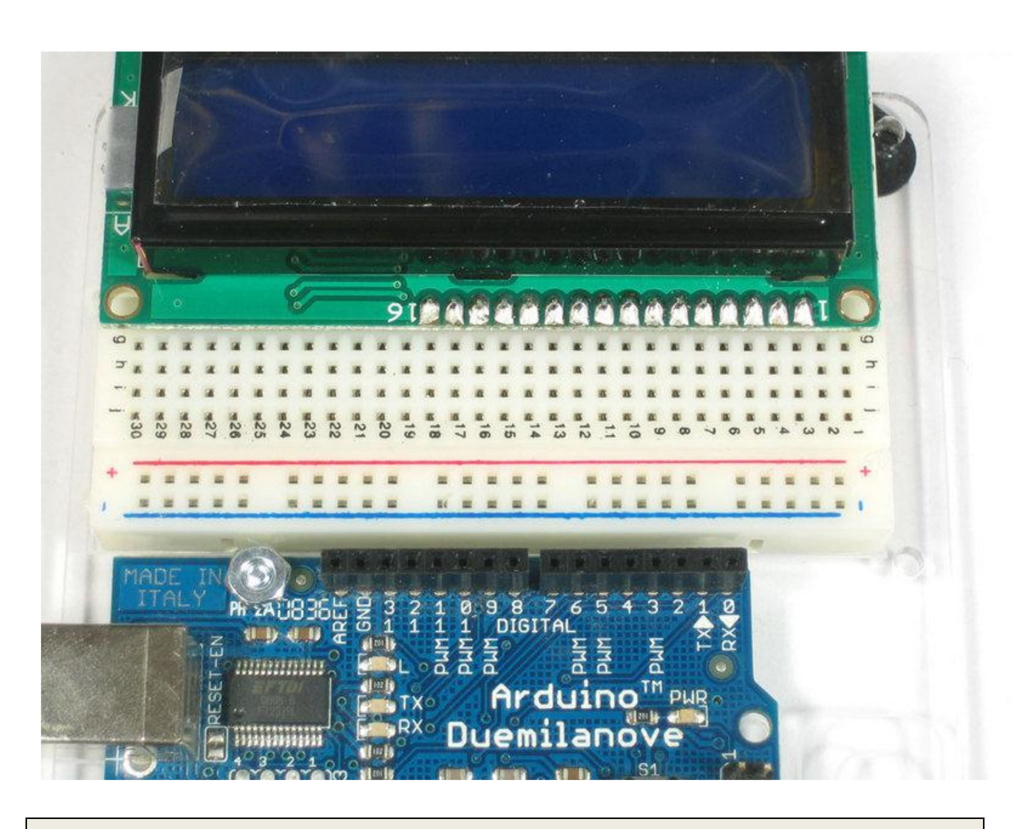


بعد ذلك قم بلحام نقطه التوصيل الاخيره وذلـك حتـى تقـوم بتثبيـت pin headers من كلا الجانبين

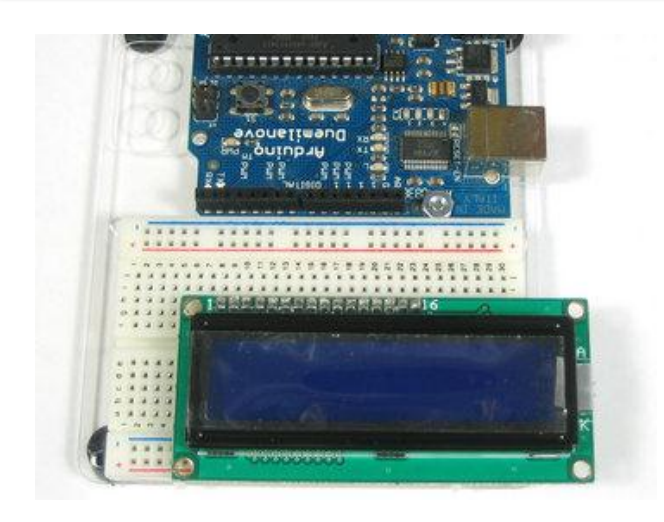
**قم بتكرار نفس العمليه السابقه مع جميع نقاط التوصيل حتى تنتهى من 16 نقطه كما فى الصوره التاليه**



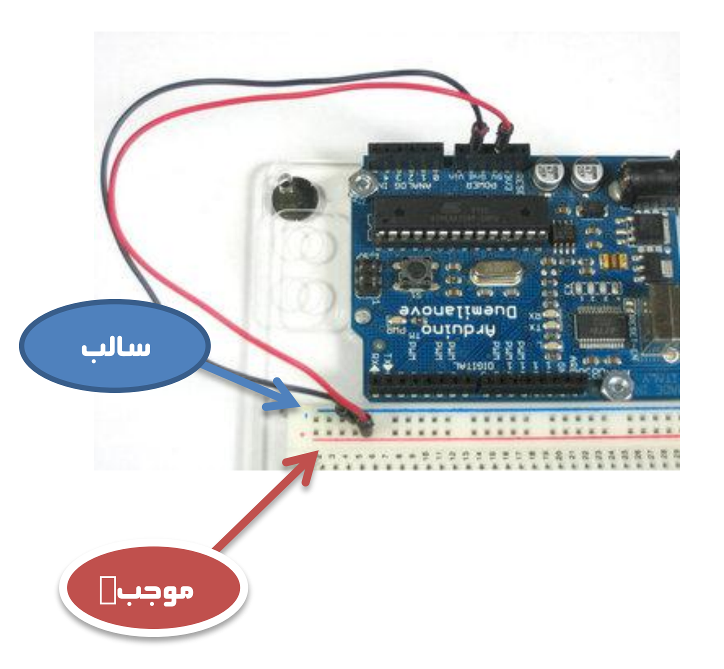
بعد الانتهاء من اللحام قم بتركيب الـ pin header على لوحه التجارب كما في الصورة



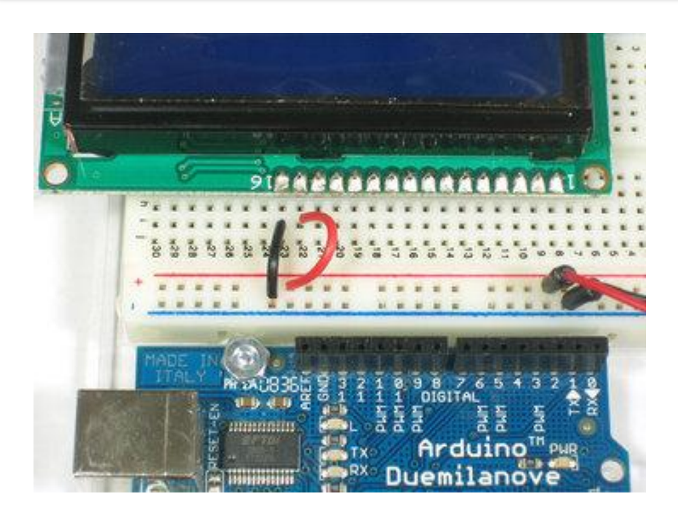

يفضل استخدام قاعده تثبيت بلاستيكيه تستخدم فى تثبيت اردوينو و لوحه التجارب معا المثل التى تستخدم فى الصوره التاليه



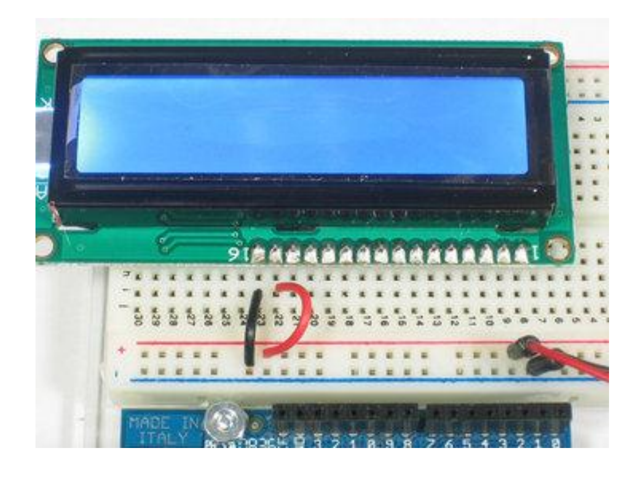
قاعده التثبيت البلاستيكيه تحمل اردوينو و لوحه التجارب و الشاشه معا




- الان نقـــــــوم بتوصيل المخرج 5v على خـط الموجـب الاحمـر و الطرف GND على الخط الارضـــى فـــى لوحــه التجارب




- بعد ذلك نقوم بتوصيل المدخل رقم 16 في شاشه العرض على الخط الارضي و نوصل المدخل رقم 15 على الخط الموجب v 5 كما في الصوره التاليه :

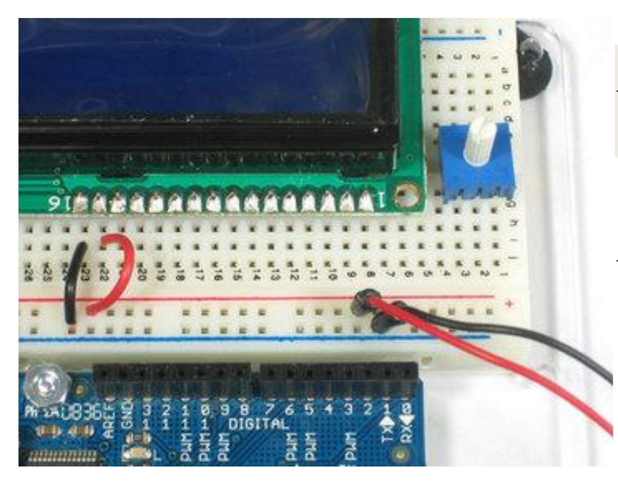



- قم بتوصيل اردوينو بالبطاريه او كابل الـ USB ثم لاحظ ان الضوء الخلفي للشاشه بدأ في العمل كما في الصوره التاليه :

ملحوظه يختلف لون الاضائه على حسب الشاشه المستخدمه و تتوفر الالوان التاليه:

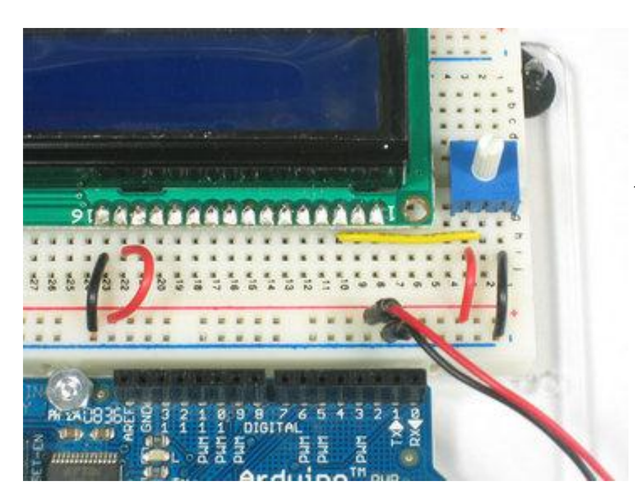
الاحمر – الأخضر – الأزرق – الأبيض –

# تركيب المقاومه المتغيرة للتحكم في شدة الاضاءة

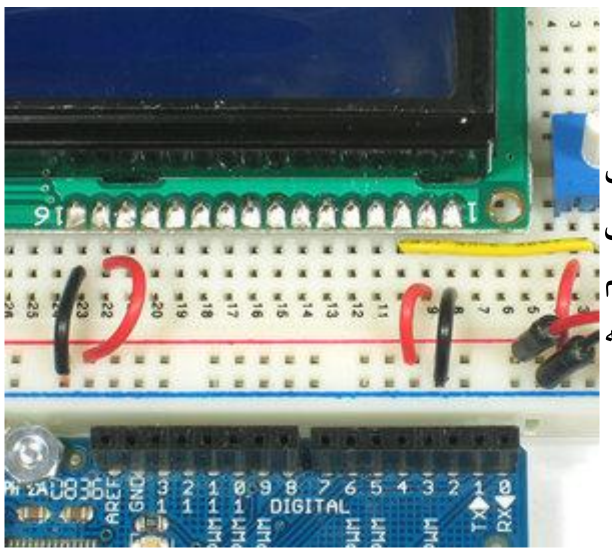



تعتبر تلك الخطوه اختياريه و يمكنك ان تهملها اذا اردت.

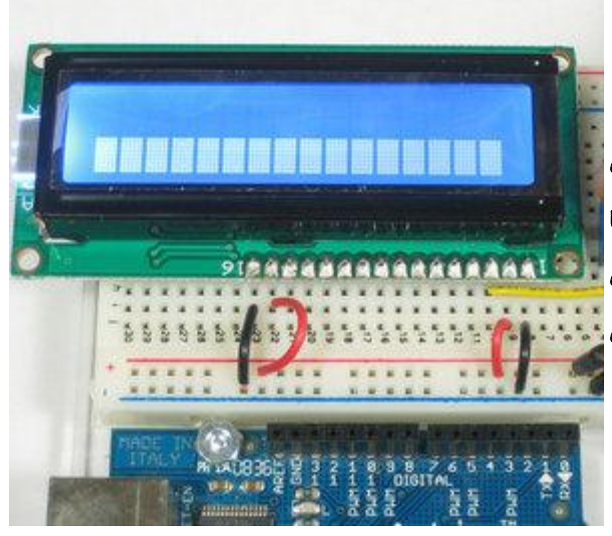
الهدف من تركيب المقاومه المتغيره هو التحكم فى التيار الكهربى الداخل الى الشاشه و بالتالى التحكم فى شده السطوع (الاضاءه) الناتجه



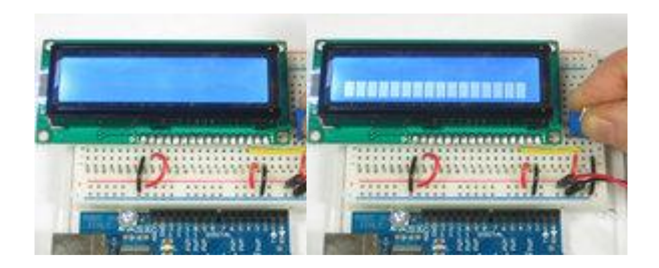

- قم بتوصيل احد الاطراف الجانبيه للمقاومه المتغيره بالخط الموجب و الطرف الجانبى الاخر بالخط السالب ثم قم بتوصيل الطرف الاوسط للمقاومه المتغيره بالمدخل رقم 3 على الشاشه كما فى الصوره التاليه :



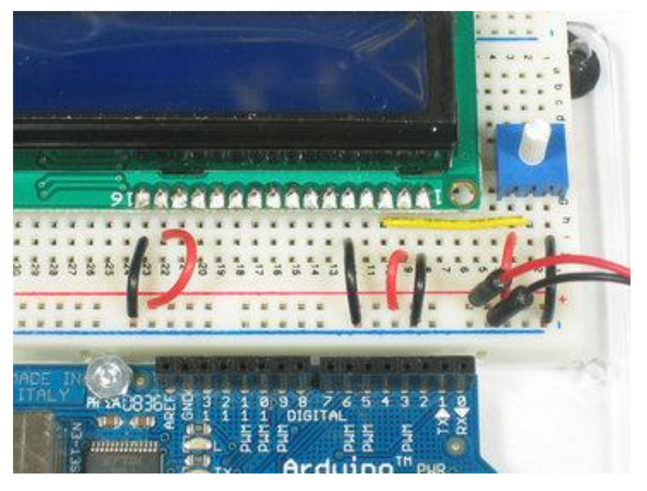

- قم بتوصيل المدخل رقم 1 على شاشه العرض بالخط الارض و المدخل رقم 2 بالخط الموجب على لوحه التجارب



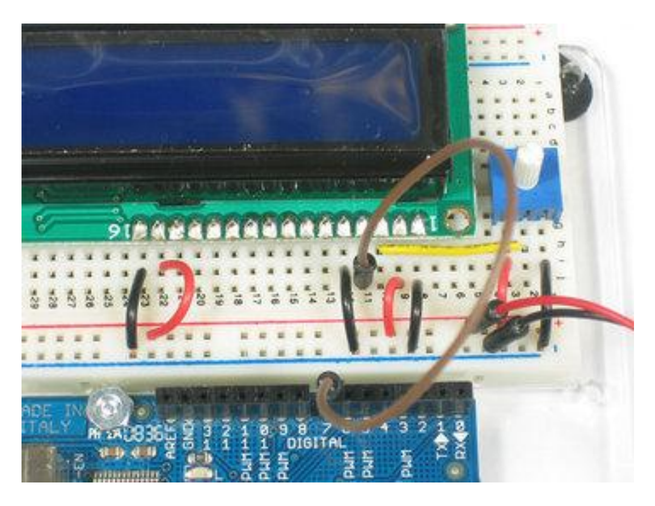

الان قم بوصيل البطاريه مره اخرى و قم بتدوير عصا الدوران فى المقاومه المتغيره ولاحظ ماذا سيحدث للاضاءه الخلفيه لشاشه العرض



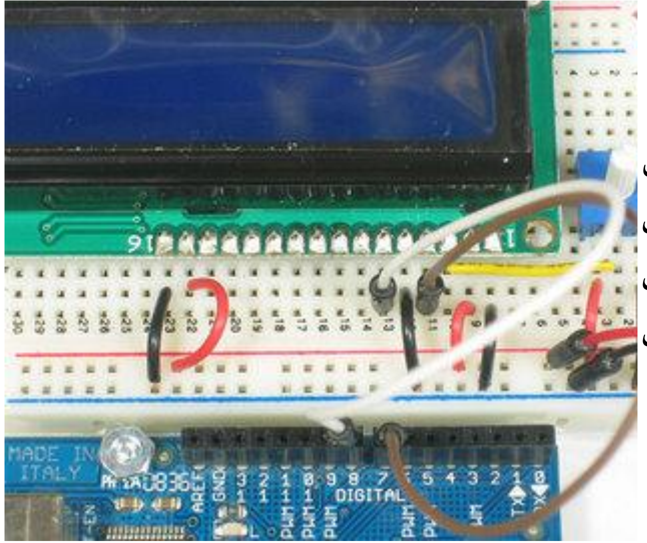
جميع الخطوات السابقه كانت تهدف لتوصيل الشاشه بالطاقه و بالمقاومه الضوئيه التى تتحكم فى مدى سطوع الشاشه و شده الاضائه الخلفيه ، الخطوات التاليه ستوضح كيف توصل منافذ نقل البيانات من اردوينو الى الشاشه.



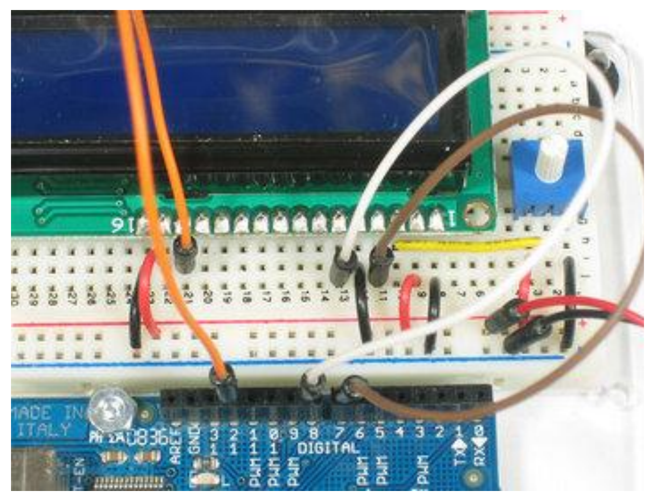

فى بعض المشاريع قد تستخدم المدخل رقم 5 و الذى يسمى RW لكن فى حالتنا هذا سنقوم بتوصيله بالخط الارضى



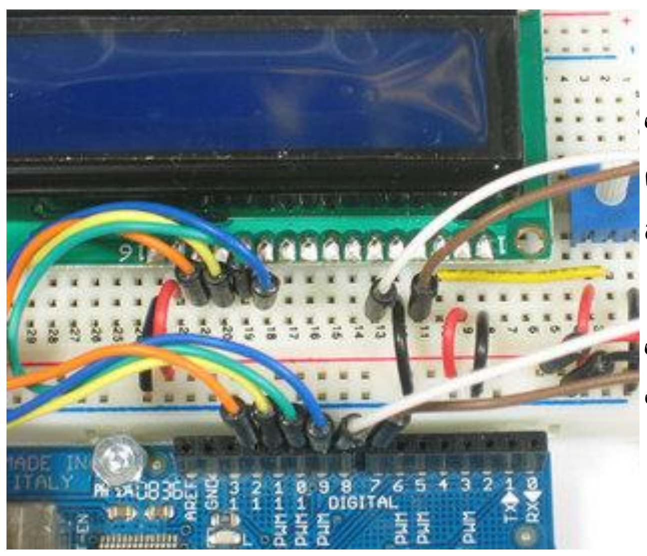
قم بتوصيل المدخل رقم 4 فى شاشه العرض على المخرج رقم 7 فى بورده اردوينو



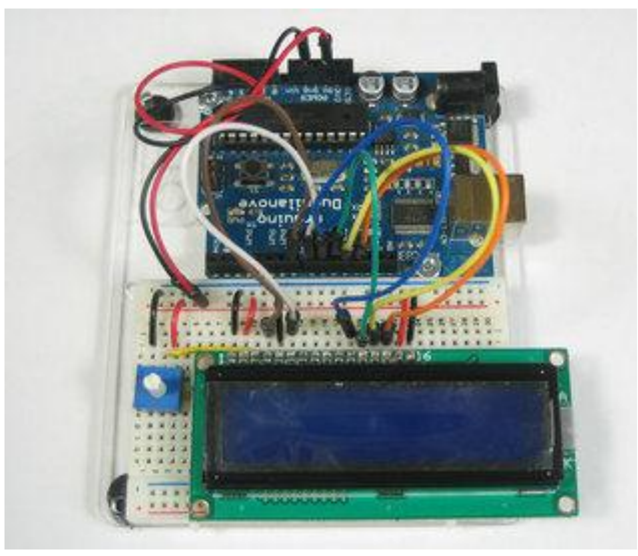
بعد ذلك نقوم بتوصيل المدخل رقم 6 فى شاشه العرض على المخرج رقم 8 فى بورده اردوينو




قم بتوصيل المدخل رقم 14 في شاشه العرض على المخرج رقم 12 في بورده اردوينو




الخطوه الاخيره هي توصيل المداخل رقم 13,12,11 بالمخارج رقم 11,10,10 الموجوده على اردوينو بنفس الترتيب كما في الصوره التاليه



الشــكل النهــائى بعــد توصيل جميع الاطراف اللازمـه لتشـغيل الشاشـه مع اردوينو.

#### الان حان وقت كتابه الكود البرمجى

توفر بيئه تطوير اردوينو العديد من الامثله الجاهزه و يمكنك ان تستخدم مثال الشاشه الجاهز من خلال فتح قائمه الامثله كتالى:

**File→Examples→LiquidCrystal→HelloWorld**

سنحتاج ان نعدل السطر البرمجى التالى :

```
LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
```

ليصبح هكذا:

```
LiquidCrystal lcd(7, 8, 9, 10, 11, 12);
```

بعد تعديل السطر قم بتأكيد و رفع الكود الى بوره اردوينو كما فعلت فى الامثله السابقه

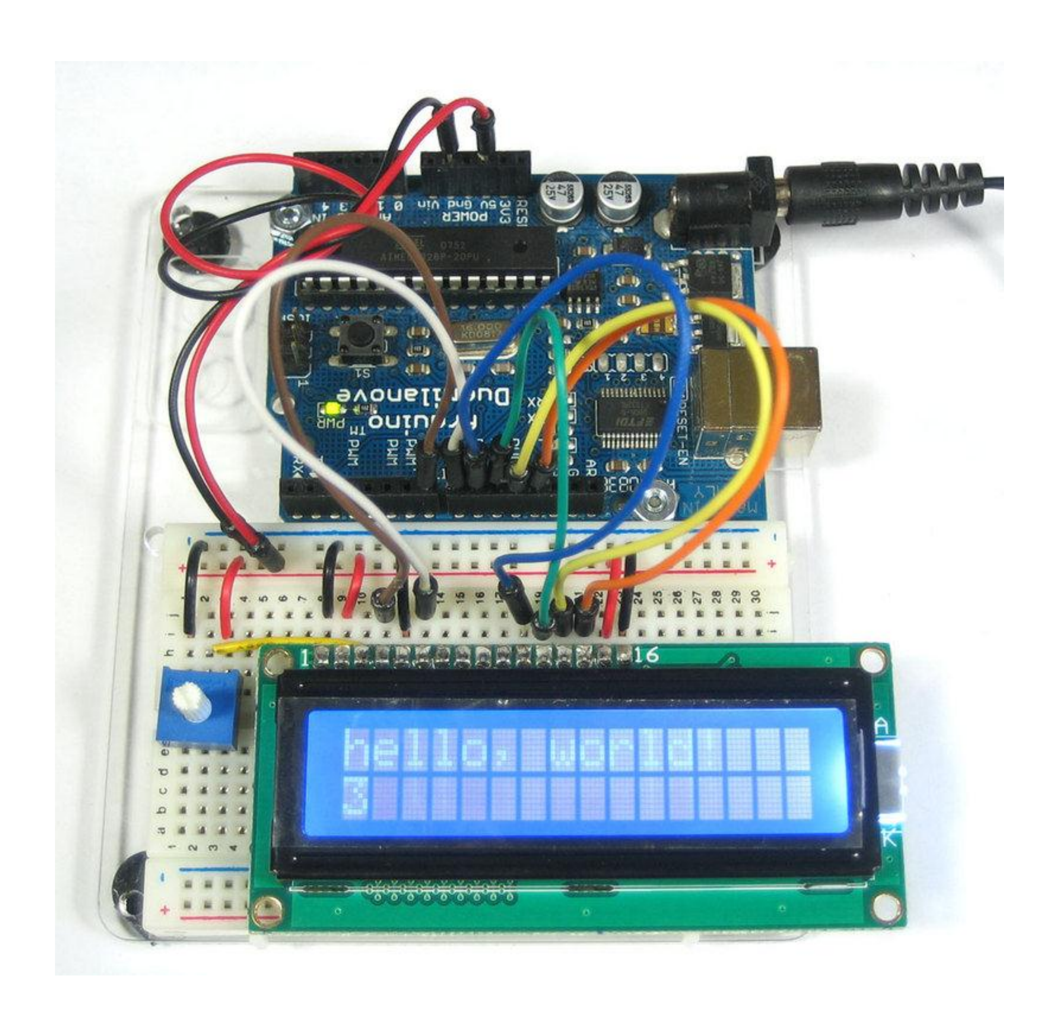
الكود البرمجي بعد التعديل:

```
//Example_12_LCD_16x2
#include <LiquidCrystal.h>
LiquidCrystal lcd(7, 8, 9, 10, 11, 12);

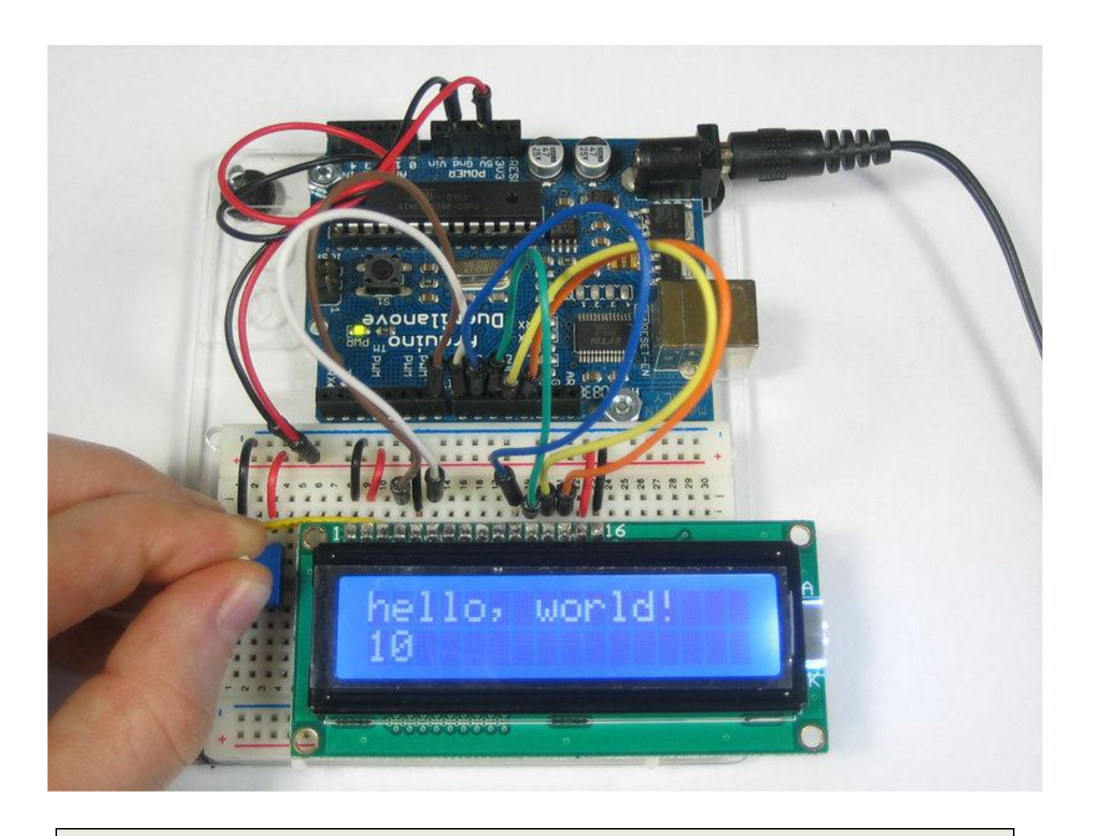

void setup()
 {
  lcd.begin(16, 2);
  lcd.print("hello, world!");
}


void loop()
 {
  lcd.setCursor(0, 1);
  lcd.print(millis()/1000);
}
```

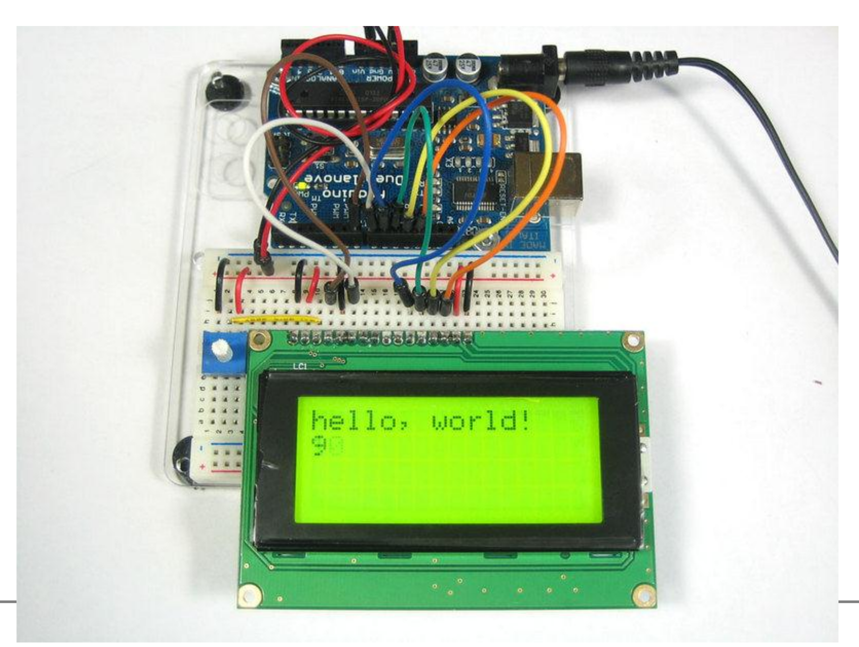
## الشكل النهائي للمثال العاشر (^_^)



تستطيع تعديل مستوى سطوع الشاشه عن طريق تغير قيمه المقاومه المتغيره




كما يمكنك استخدام اى حجم و لون تفضله من الشاشات Character LCD مثل الحجم الاكبر فى عدد الحروف 20x4 كما فى الصور التاليه

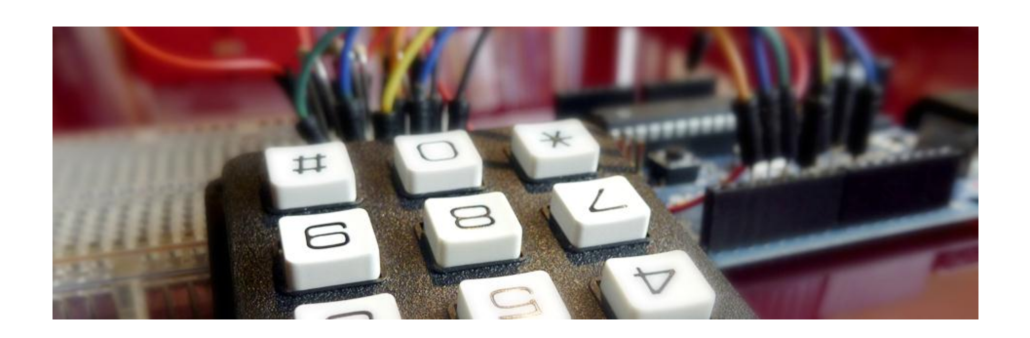

hello, world!
90




## مصادر اضافيه للمعلومات:

لمزيد من المعلومات و الأوامر المستخدمه مع شاشات LCD
http://arduino.cc/en/Tutorial/LiquidCrystal
لعمل اشكال مرسومه على شاشه من نوع character LCD
http://www.instructables.com/id/Controlling-a-character-LCD-with-an-Arduino
لتوصيل شاشه هواتف نوكيا الملونه Nokia 6100 LCD او Nokia 2600
http://www.instructables.com/id/How-To-Use-a-Nokia-Color-LCD/

# استخدام لوحه الأرقام Keypad مع اردوينو



تعتبر لوحه الارقام من اهم وسائل الادخال المستخدمه بكثره فى مشاريع المتحكمات الدقيقه و التى ستجدها حولك فى العديد من الاجهزه الالكترونيه مثل الهاتف المنزلى و لوحه المفاتيح و لوحه التحكم فى المصاعد الموجوده فى البيوت متعدده الطوابق ... الخ

و تختلف اشكال لوحات الارقام تبعاً لحجمها و عدد الارقام المتاحه ، فى بعض اللوحات ستجد رموز اضافيه مثل علامه النجمه (*) و علامه الشباك (#) او حتى بعض الحروف الانجليزيه مثل A,D,F و تعتبر لوحه الارقام بحجم 4x3 و 4x4 هى اشهر لوحات الكتابه




3x4

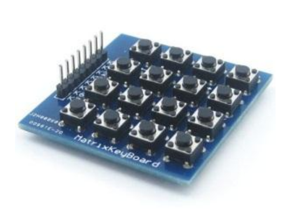



4x4

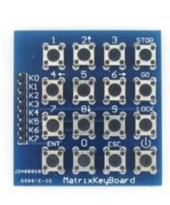





4x4






كما تتوفر بعض اللوحات المرنه و القابله للطى بسهوله و التى تصنع من نوع خاص من البلاستيك المرن و تتميز بالنحافه حتى تبلغ ثخانه اللوحه بضعه مليمترات فقط و تتميز بالسعر المنخفض (حوالى 2 دولار فقط )






سوف نشرح لوحه الارقام المرنه بمقاس 3x4 و التى تتميز بالمواصفات التاليه:

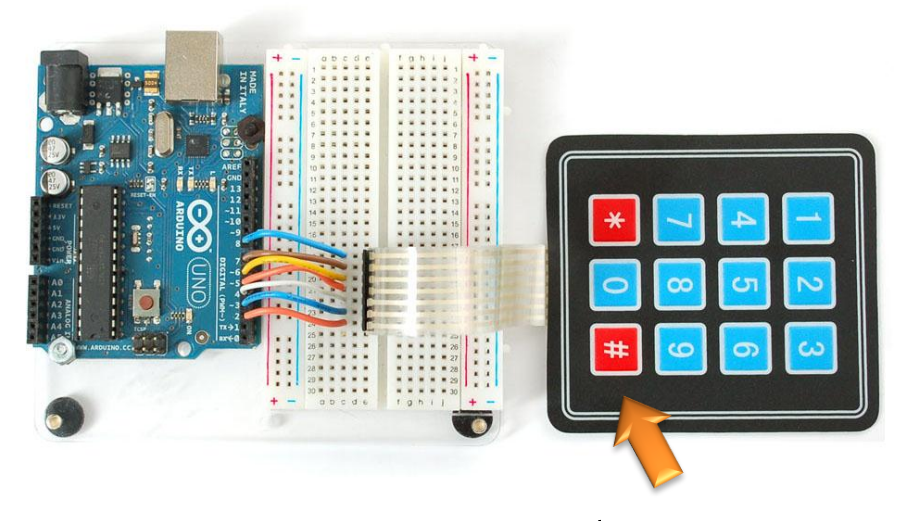
- Weight: 7.5 grams
- Keypad dimensions: 70mm x 77mm x 1mm (2.75" x 3" x 0.035")
- Length of cable + connector: 85mm
- 7-pin 0.1" pitch connector

# المثال الحادى عشر أستخدام لوحه ارقام




## مكونات المثال الحادى عشر:

- لوحه ارقام (مرنه) بمقاس 3x4
- أعمده توصيل نحاسيه 7 نقاط ( 7) Pin-Headers
- اردوينو Arduino Uno
- لوحه تجارب Breadboard

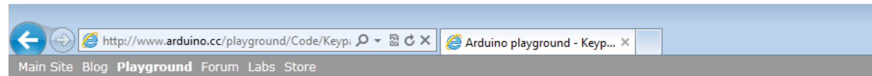


قم بتوصيل مخارج اردوينو بدئاً من المخرج رقم 2 حتى المخرج رقم 8 بلوحه الأرقام مع ملاحظه ان يكون رمز # هو الرمز المتواجد امام المخرج رقم 2 فى بورده اردوينو كما فى الصوره بالظبط و بذلك نكون قد انتهينا من تجهيز لوحه الأرقام.

قبل ان نبدأ كتابه الكود البرمجى سيتوجب تحميل مكتبه لوحه الارقام من موقع اردوينو حيث لا تتوفر هذه المكتبه بشكل افتراضى فى برنامج Arduino IDE على عكس مكتبه استخدام الشاشات من نوع Character LCD و التى تتوفر بشكل افتراضى داخل برنامج Arduino IDE

أولاً: توجهه الى اللينك التالى على موقع اردوينو الرسمى:
http://www.arduino.cc/playground/Code/Keypad

ثم قم بتحميل مكتبه لوحه الارقام كما فى الشكل التالى:




Version 3.0 has just been posted (19 July 2012) and was rewritten to support multi-keypresses by default. But for those who still need the original single-keypress functionality, the library is fully backwards compatible.

You won't need external resistors or diodes because the library uses the internal pullup resistors and additonally ensures that all unused column pins are high-impedance.

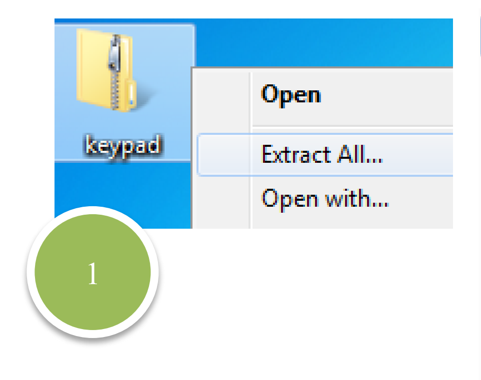
Support was added to allow other hardware to be used with the keypad. When you download the Keypad library you will also get the Keypad_I2C library. That library adds support for using a matrix keypad with I2C port expanders. You can find more information inside the docs folder under the Keypad_I2C library folder. Start with the noHeaders example sketch to learn which libraries are needed.

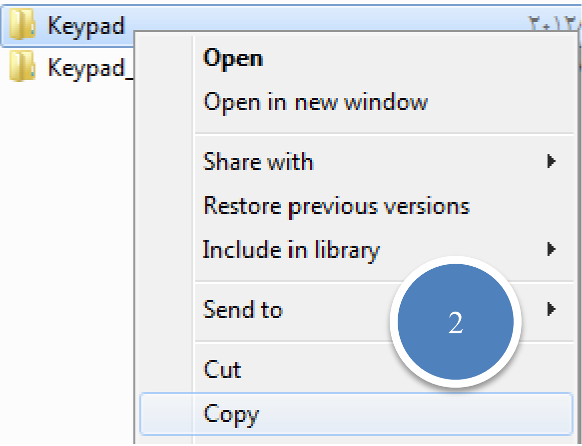
## Download, install and import

Download here: keypad.zip

Put the Keypad folder in "*arduino*\libraries\".
In the Arduino IDE, create a new sketch (or open one) and select from the menubar "Sketch -> Import Library -> Keypad".
Once the library is imported, an "#include <Keypad.h>" line will appear at the top of your Sketch.

بعدها سيبدأ المتصفح بتنزيل مكتبه لوحه الأرقام و هى عباره عن ملف مضغوط اسمه keypad.zip ، الخطوه التاليه هى ان تقوم بفك ضغط الملف و نسخ الفلودر المسمى keypad و توجهه الى مكان تواجد برنامج اردوينو **Arduino IDE** حيث ستجد فولدر اسمه libraries قم بفتح الفولدر و الصق keypad داخله كما فى الصوره التاليه:


keypad
Open
Extract All...
Open with...
1


Keypad
Keypad_
Open
Open in new window
Share with
Restore previous versions
Include in library
Send to
Cut
Copy
2

arduino-1.0.1
Include in library
Share with
Burn
New folder
Name
Date modified
Type
3
drivers
examples
hardware
java
lib
libraries
reference
tools
arduino
cygiconv-2.dll
cygwin1.dll
libusb0.dll
revisions
rxtxSerial.dll
File folder
Open
Open in new window
Share with
Restore previous versions
Include in library
Send to
Cut
Copy
Paste

بعد الانتهاء من تجهيز المكتبه البرمجيه نبدأ فى كتابه الكود:

```
//Example_13_Keypad_Input
#include <Keypad.h>
const byte ROWS = 4;
const byte COLS = 3;
char keys[ROWS][COLS] =
{
  {'1','2','3'},
  {'4','5','6'},
  {'7','8','9'},
  {'#','0','*'}
};
byte rowPins[ROWS] = {5, 4, 3, 2};
byte colPins[COLS] = {8, 7, 6};


Keypad keypad = Keypad( makeKeymap(keys), rowPins, colPins, ROWS,
COLS );

void setup()
{
  Serial.begin(9600);
}


void loop()
{
  char key = keypad.getKey();


  if (key != NO_KEY) { Serial.println(key); }
}
```

عدد الصفوف

الحروف و الرموز

أمر يقوم بتفعيل استقبال الارقام من لوحه الارقام

بعد الانتهاء من كتابه الكود قم بتشغيل serial monitor و اضغط على الأرقام فى الـKeypad و انظر ماذا سيحدث !!

## مصادر اضافيه للمعلومات:

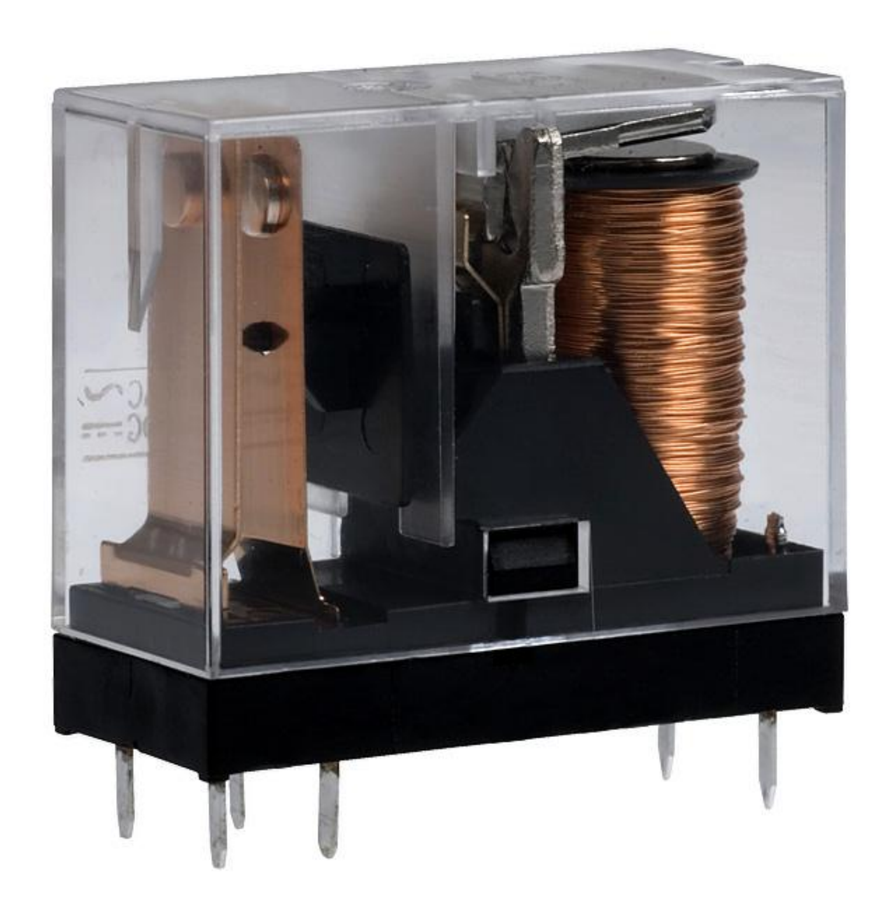
مكتبه لوحه الأرقام و الأوامر البرمجيه :
http://www.arduino.cc/playground/Code/Keypad

كيف تصنع قفل الكترونى بأستخدام اردوينو و السيرفو موتور:
http://www.instructables.com/id/Access-control-with-Arduino-Keypad-4x4-Servo

كيف توصل لوحه الأرقام بأردوينو عن طريق 3 أسلاك فقط بدلا من 7 أسلاك:
http://www.instructables.com/id/Arduino-3-wire-Matrix-Keypad

عمل نظام امنى (مفتاح الكترونى شامل مكون من اردوينو ميجا و شاشه و لوحه ارقام و قفل الكترونى (سيرفو موتر):
http://www.instructables.com/id/Password-Lock-with-Arduino

# أستخدام المُرحِل Relay مع اردوينو



يعتبر الريلاى من اهم العناصر المستخدمه فى التحكم الالكترونيى بأنواعه المختلفه

## ما هو ذلك العنصر ؟ وكيف يمكن استخدامه؟ وماهي تطبيقاته؟ إيجابياته؟ سلبياته؟

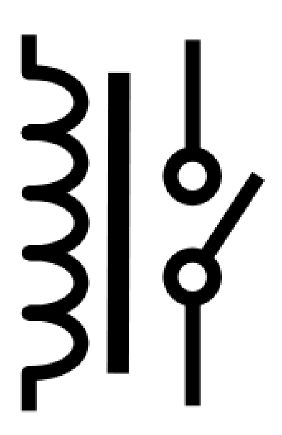
### مم يتكون؟

الريلاي هو عنصر ميكانيكي/إلكتروني، ويمكننا تخيله على شكل مفتاح أو زر كهربائي، داخلياً يتكون من جزئين رئيسيين:

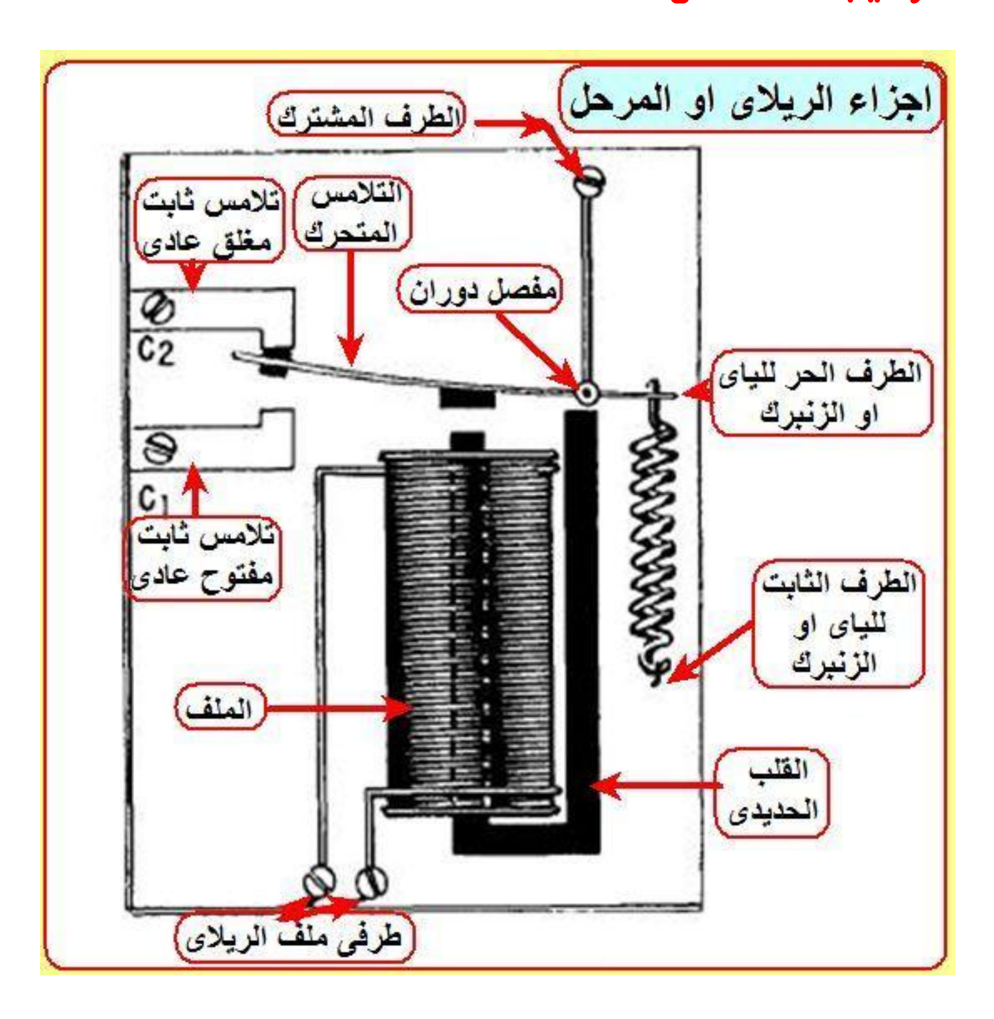
**الأول:** سلك ملفوف حول قالب حديدي، وفي مقدمة ذلك القالب يتوضع **الجـزء الثـاني:** وهو لسان أو قطعة معدنية مستطيلة الشكل تقع أمام القالب/الملف وهي بمثابة المفتاح فهي تحتوي على تماسات معدنية يتم من خلالها توصيل حمل كهربائي ليعمل الريلاي على فصله أو تشغيله بحسب وضع الملف في الريلاي.

### ماهو الرمز الإلكتروني للريلاي في الدارات الإلكترونية؟

يرمز للريلاي بالشكل التالي: حيث الجزء الأيسر هو الملف والأيمن هو رمز المفتاح الكهربائي



## التركيب من الداخل




## كيف يعمل؟

عندما يتم تطبيق فرق جهد على طرفي الملف الداخلي للريلاي، سيمر تيار في الملف ليتحول بواسطته الملف لمغناطيس كهربائي مولدا مجالا مغناطيسيا سيقوم بدوره بجذب

ذلك اللسان أو القطعة المواجهة للملف بحيث تغلق التماسات الكهربائية، وعند فصل الجهد المطبق سيتلاشى التيار تدريجيا ليختفي ذلك المجال المغناطيسي، وهناك زنبرك سيقوم بإعادة اللسان لوضعه الطبيعي وفصل التماسات وفتح المفتاح الذي أغلق.

إذا بكل بساطة فالريلاي هو مفتاح كهربائي ميكانيكي، ما إن يتم تطبيق جهد على ملفه، يبدأ بالعمل ليجذب تماسات معدنية ستعمل على غلق أو فتح دارة خارجية متصلة بها.

## ومافائدته إذا؟ ألا يمكننا استخدام مفتاح كهربائي مباشرة؟




تخيل معي أننا نحتاج ان نوصل اردوينو بحمل كهربائي بعمل على فرق جهد متردد قيمته 110 فولت او 220 فولت. منطقيا، لا يمكننا تشغيل ذلك الحمل من خلال اردوينو لأن اقصى جهد يمكن توليده من اردوينو هو 5 فولت فقط ،

وهنا يأتي دور الريلاي فهو سيقوم بعملية تشغيل لأي حمل ذا استهلاك كبير للطاقه من خلال فرق جهد صغير، والسر يمكن وراء تشغيل الملف بجهد صغير مثل 5فولت لنقوم بتشغيل المصباح الذي سيتم توصيله على أطراف التماسات التي سيجذبها الملف داخل الريلاي.

**إذا للريلاي** فائدة عظيمة كونه سيعمل على ترحيل جهد وتيار صغير لقيادة حمل كبير. وأيضا ميزة الريلاي تكمن في قدرته على العزل التام ما بين دارة المصدر والحمل "الخرج"، فهما معزولان تماما ولكل واحد منهما نقطة أرضية مرجعية مختلفة عن الأخرى،

وهذا له فائدة كبيرة في منع انتقال التشويش والجهود العابرة لدارتنا وبمعنى أصح حمايتها. أخيرا، يمكن استخدام الريلاي في التحكم عن بعد، أي وضع الريلاي بالقرب من الجهاز ومد سلك للدراة التي تصدر الأوامر.

## عيوبه:

لكل عنصر إلكتروني فوائد وسلبيات ، وكذلك الأمر بالنسبة للريلاي، فأهم سلبياته هي:

- بما أنه يعتمد على الملف –وهو حثي– فلذلك ستتولد فيه قوة دافعة كهربائية عكسية عند مرور تيار به ( تعرف بقاعدة لنز ) عند التوصيل وعند الفصل مما قد يؤدي لعطب الدارة المتصلة به. إلا أنه يمكننا تجاوز هذه المشكلة البسيطة بواسطة وضع عنصر الدايود بين طرفي الملف وذلك لمنع عودة تلك القوة الدافعة العكسية .
- حدوث ارتدادات ميكانيكية عند كل تحويل من وضع الفصل الى وضع التوصيل أو العكس. مما قد يؤدي لعطب الحمل
- يحتاج إلي دائرة موائمة لكى يعمل جيدا مع الأنظمة الإلكترونية . وعادة تكون هذه الدارة هي مكونة من ترانزستور
- العمر الافتراضى للريلاي صغير نسبيا خصوصا فى الدوائر التى تتطلب عددا كبيرا من مرات الوصل والفصل وعادة ما تذكر قيمة عدد مرات الفصل والوصل القصوى لكل ريلاي في ملف المواصفات الفنية

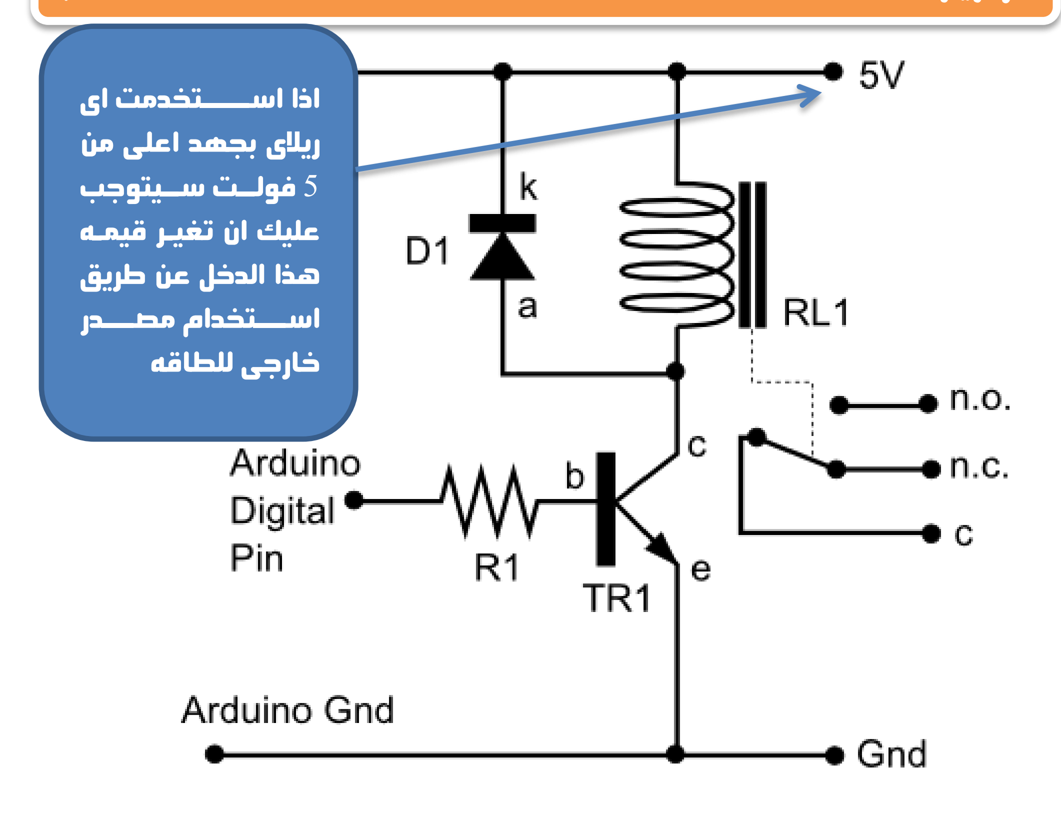
## دارة عملية لاستخدام الريلاي مع الدارت الإلكترونية

عند اختيار أي ريلاي فلابد من الانتباه للأمور التالية كي نتمكن من الاختيار العلمي الصحيح ومعرفة البدائل التي يمكن استخدامها في حالة عدم توفر الريلاي المناسب

- فرق الجهد اللازم لتشغيل الملف الداخلي للريلاي: فمثلا لو كان لدينا دارة تعمل على فرق جهد 5 فولت فيجب اختيار ريلاي له فرق جهد لملفه هو 5 فولت، وفي حالة تعذر ذلك يمكننا استخدام جهد أكبر بعد استخدام دارة مواءمة سبق أن طرحتها في تدوينة الترانزستور.
- أقصى تيار لازم لتشغيل ملف الريلاي: وغالبا هذه القيمة لا تعطى بشكل واضح بل يتم ذكر مقاومة الملف بدلا منها، ومن خلال قسمة فرق جهد الملف على مقاومته يتم حساب التيار الأعظم. وهي قيمة مهمة جدا كي نعرف هل يمكننا تشغيل ووصل الريلاي بدارتنا مباشرة أو لابد من وضع دارة مواءمة
- جهد وتيار مرحلة الخرج أو التماسات، وهنا يجب معرفة ما يحتاجه الحمل ومقارنته بالريلاي وينبغي دائما أن يتم اختيار قيمة تيار عظمى تزيد عن القيمة المطلوبة بنسبة 5 إلى 10%
- العمر الافتراضي لتماسات الريلاي وعادة ما تعطى بملايين المرات

### ننتقل الآن للدارة العملية  سنحتاج لهذه الدائره المكونات التاليه:

- ريلاى يعمل على جهد 5 فولت (يمكن استخدام 9 او 12 فولت)
- دايود
- مقاومه 1 كيلو اوم
- ترانزستور من نوع 2N2222




TR1 - BC548, 2N2222, 2N3704 transistor
D1 - 1N4004, 1N4007 diode
R1 - 1k Ω resistor
RL1 - 5V relay

c - collector
b - base
e - emitter

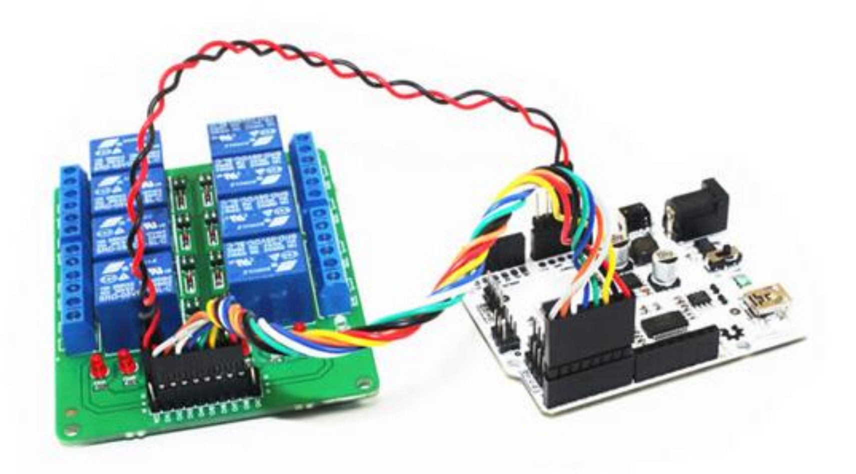
n.c. - normally closed
n.o. - normally open
c - common

a - anode
k - cathode

تتوفر فى الأسواق العديد من دوائر المُرحِلات الجاهزه للأستخدام Relay Boards مع اردوينو كما تتوفر بأحجام و اعداد مُرحِلات مختلفه مثل الصور التاليه:







في حال كان مشروعك يحتاج تشغيل احمال بفرق جهد كبير مثل 220 فولت فانني انصح بشده بأن تشترى بورده جاهزه بالحجم المناسب للمشروع الخاص بك و ذلك لأن هذه البوردات تصنع بجوده و دقه عاليه و هو ما تحتاجه في المشاريع التي تعمل بجهد كبير تجنباً لحدوث اى اخطاء قد لا يُحمد عقباها.

## مصادر اضافيه للمعلومات:

هناك مشروع جميل يشرح كيفيه استخدام الريلاى مع اردوينو لتجهيز منفذ للطاقه بجهد 220 فولت يعمل من خلال اشارات التحكم الصاده من اردوينو
http://www.sparkfun.com/tutorials/119

ملحوظه: مرجع الصور و المعلومات المذكوره عن المُرحِل **relay** في هذا الفصل

http://muslimlead.com/?p=2121

و تحتوى المدونه ايضا على العديد من الدروس الرائعه حول نظريات عمل العديد من المكونات الالكترونيه مثل الدايود، الثنائي المشع للضوء، المكثف، الملف، الترانزستور

يمكنك الضغط على اى اسم من اسماء العناصر المذكوره ليتم نقلك الى موضوع يشرح اساسيات العنصر بالتفصيل ☺

## ملاحظات شخصيه :

هذه الصفحه مخصصه لكتابه ملاحظاتك الشخصيه عن الفصل السابع :



# من أقوال العظماء:

سُبحانكَ اللهم خيــرَ معلـــــم

علّمتِ بالقلمِ القرونَ الاولــى

أخرجتَ هـذا العقل من ظُلُماتـهِ

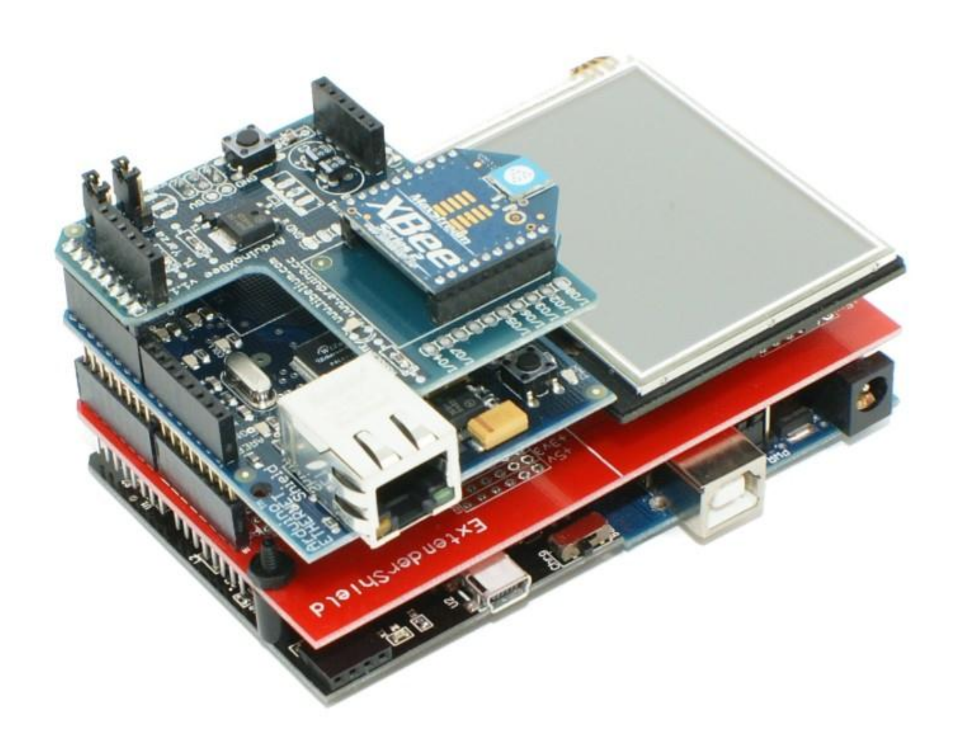
وهَديتهُ النـــورَ المبيـنَ ســـبيلا

أمير الشعراء أحمد شوقى

# الفصل الثامن

## أغطيه اردوينو الجاهزة

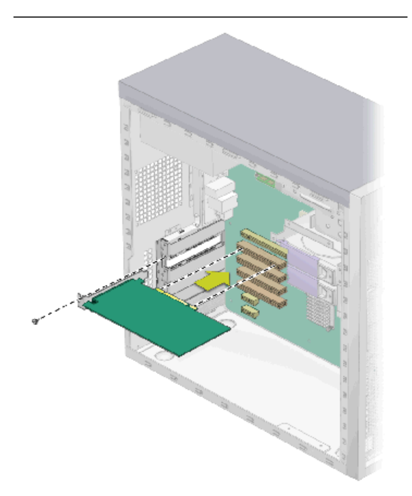
## Arduino Shields

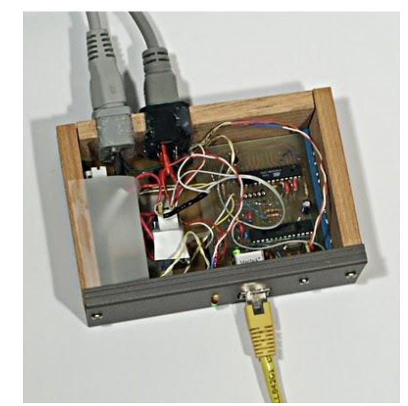


**من اكثر المميزات** المثيره في اردوينو هي توافر عدد كبير من الأغطية التي يمكن تركيبها مباشره على البوردات المختلفة مثل Arduino uno و الإصدارة الأعلى في الإمكانيات و قوه المعالجة Arduino Mega

---

تعمل الاغطيه بنفس فكره كروت PCI الموجوده فى الحاسب الآلى ، على سبيل المثال اذا لم توفر اللوحه الام Motherboard كارت شبكه للدخول على الانترنت فأنك تستطيع بكل بساطه ان تشترى كارت شبكه network interface card و تقوم بتركيبه و الدخول على الانترنت فورا .. هكذا تعمل اغطيه اردوينو





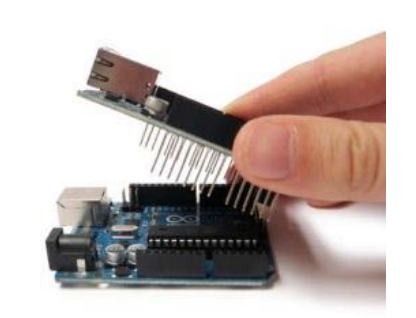
اذا كان لديك خبره سابقه مع المتحكمات الدقيقه فلابد انك تدرك مدى الصعوبه و المجهود المبذول حتى تقوم بتوصيل المتحكم الدقيق بالانترنت او شبكه محليه (حيث ستقوم ببناء وحده Ethernet من الصفر بنفسك) و سيستهلك ذلك الكثير من الوقت و المجهود ، اما فى حاله اردوينو كل ما عليك فعله ان تشترى الغطاء جاهز و تركبه مباشره على البورده الخاصه بك و تبدأ فى برمجه مشروعك بكل سهوله

**تتوفر العديد من الاغطيه و سنستعرض منها التالي:**

## • غطاء اردوينو للشبكة السلكية Arduino Ethernet Shield



**يـوفر** ذلـك الغطـاء امكانيـه توصـيل اردوينـو بشـبكات الحاسـب الآلي و الانترنت عن طريق توصيل كابل شبكه من نـوع CAT5 مثل المستخدم فـي الحاسب الآلي و يمكنك استغلال ذلك الغطاء في عمل مشاريع التحكم عن بعـد بأسـتخدام الشـبكه المحليـه او الانترنـت فيمكنـك مـثلا توصـيل عـدد مـن الاجهزه الالكترونيه بالادروينو و التحكم بها من اى مكان فى العالم عن طريق الانترنت ، و هناك بعض الاصدارات الحديثه من هذا الغطاء توفر ايضا امكانيه تركيب كارت ذاكره لتخزين الملفات و يمكن استغلال ذلك في بعض المشاريع التى تحتاج تسجيل قيم بصوره دوريه مثل مشروع لتسجيل درجات الحراره كل فتره زمنيه معينه و تخزين القيم على كرت ذاكـره ثـم ارسالها عـن طريـق الانترنـت الى صاحب المشروع

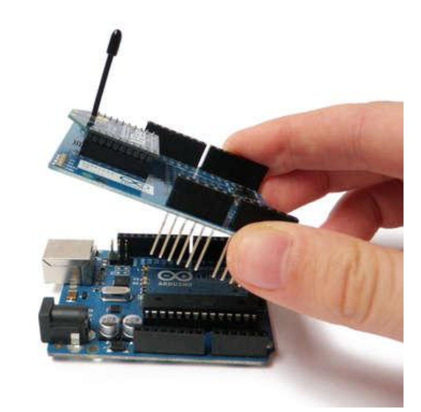




## • غطاء الشبكات اللاسلكية Xbee Shield




يـوفر هـذا الغطـاء نفـس الوظـائف الموجـوده فـى Ethernet Shield لاكـن بصـوره لا سلـيكه Wireless حيث يمكنك من توصيل الاردوينو باى شبكه لاسليكه فى نطاق يقارب 100 متر (يختلف النطاق بناء على قوه الاشاره اللاسلكيه)



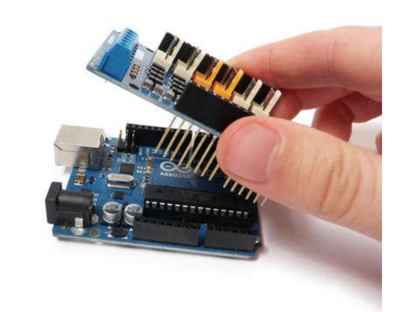




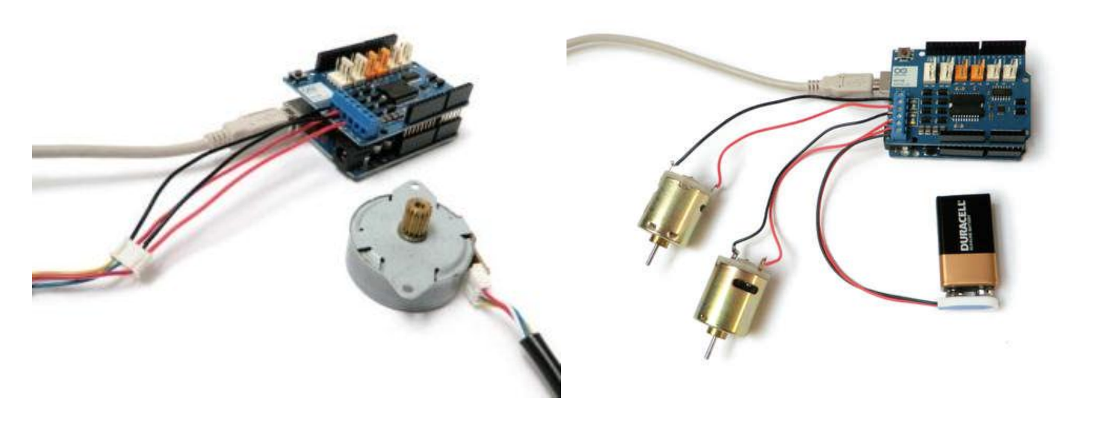

## • غطاء المحركات (الموتور) Arduino Motor Shield



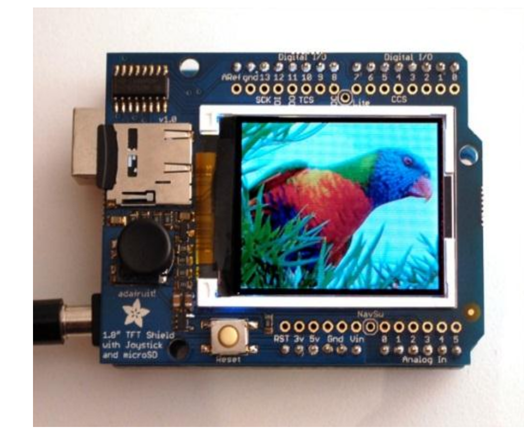
يستخدم هذا الغطاء في توصيل المحركات من نوع DC Motor, Servo Motors, Stepper Motors حيث يمكنك توصيل 3 محركات في نفس الوقت (ملحوظه: بعض



اصدارت هذا الغطاء تسمح بتوصيل محركين فقط) و يمكنك استغلال هذا الغطاء في المشاريع التي تحتاج محركات مثل مشاريع الإنسان الآلي Robots و ماكينات CNC

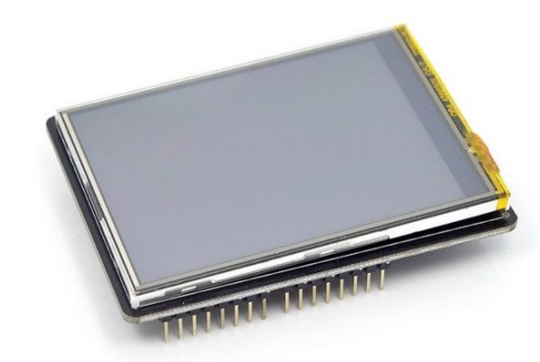


- **غطاء الشاشه الملونه التى تعمل باللمس** Arduino colored Touch screen



يستخدم ذلك الغطاء في توفير شاشه ملونه تفاعليه تعمل بالاستجابة للمسات الأصابع و يمكن استخدامها في المشاريع التي تحتاج إلى وسيله لعرض صوره او بيانات معقده و التي لا تكفى الشاشات lcd ذات اللون الواحد لعرضها، و يتوفر ذلك الغطاء بأحجام و مقاسات مختلفة تبدأ من مساحه 2 انش مربع إلى 4 انش مربع (الإنش هو وحده قياس تساوى 2.5 سنتي متر)





هناك قائمه كبيره بأسماء و مواصفات أغطيه اردوينو المختلفة تجدها على الموقع التالي:
http://shieldlist.org

## ملاحظات شخصيه :

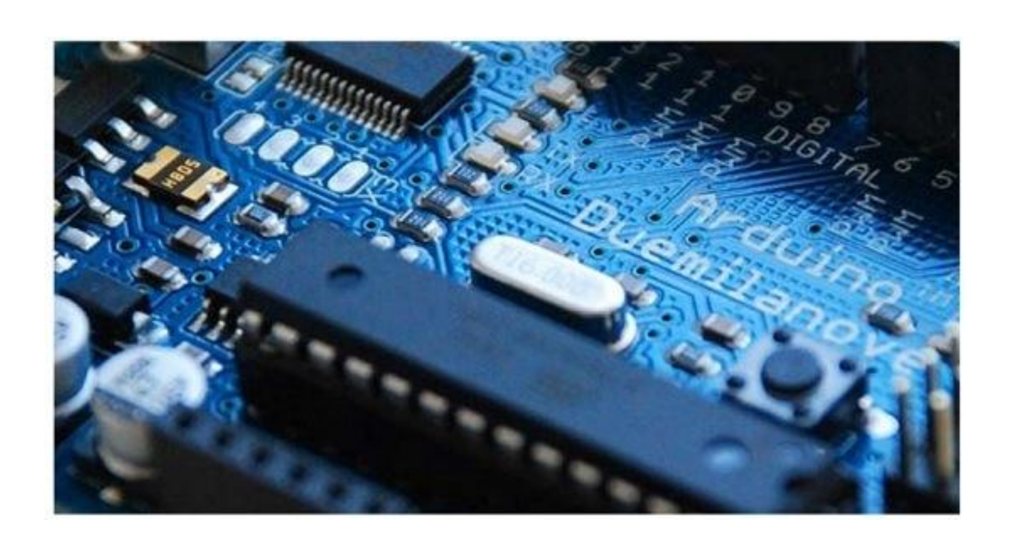
هذه الصفحه مخصصه لكتابه ملاحظاتك الشخصيه عن الفصل الثالث :



# الفصل التاسع

## اردوينو و تطور ثورة المتحكمات الدقيقه مفتوحه المصدر من

## عائله AVR الى ARM Cortex انتهائاً بـFPGA



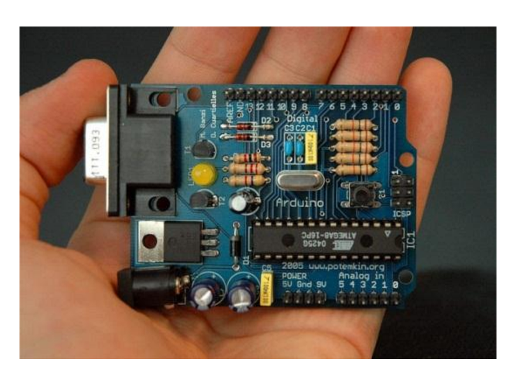
**تبدأ** القصه فى عام 2005 فى مدينه ايفريا **Ivrea** الايطاليه عندما أجتمع فريق من مهندسي الألكترونيات ليناقشوا سُبل تعليم طُلاب الجامعه و المعاهد التقنيه علوم المتحكمات الدقيقه او ما يعرف بأسم الـ Microcontrollers ، كان الهدف من اللقاء ايجاد أسلوب سهل و غير مكلف لأدخال هذا النوع من العلوم فى الدراسه الجامعيه و يكون متاح للهواه فى ذات الوقت دون ان تكون هناك قيود أو تراخيص مكلفه يحتاجون لشرائها لأستخدام البرمجيات التى كانت تستخدم فى برمجه المتحكمات الدقيقه فى هذا الوقت .. من هنا تبدأ رحله نشأه اردوينو Arduino



قام كل من " ماسيمو بانزى Massimo Banzi " بالتعاون مع " دايفيد كوارتيليس David Cuartielles و جاينلوكا مارتينو Gianluca Martino " بأطلاق مشروع أردوين ايفريا **Arduin of Ivrea** و تمت تسميه المشروع بأسم اشهر شخصيه تاريخيه فى المدينه و كان الهدف الأساسى للمشروع هو عمل بيئه تطوير للمتحكمات دقيقه بصوره مفتوحه المصدر 100 فى المئه و تضمن هذا المشروع عمل بيئه تطوير برمجيه للمتحكمات الدقيقه Integrated Development Environment و تكون مجانيه فى ذات الوقت كما تضمن عمل لوحات تطوير Development Boards صغيره الحجم بتكلفه بسيطه يمكن للطلاب و الهواه التقنين تحمل سعرها.

## Wiring

**بدأ تطوير** المشروع اعتماداً على مشروع سابق اسمه Wiring Platform و الذى قد نجح فى تحقيق بعض أهداف مشروع اردوينو الأساسيه مثل المصدر المفتوح لكن لم يصل الى المستوى الذى تمناه فريق التطوير من الناحيه البرمجيه و التصميم العتادى Hardware Design لذلك تم البدأ فى العمل على تطوير اللغه البرمجيه المستخدمه فى مشروع Wiring و جعلها أسهل و أفضل كما تم دمج بعض التقنيات المستخدمه فى لغه Processing و سميت هذه اللغه الجديده بأسم **Arduino C** و هى مشابهه جداً للغه البرمجيه C++ من ناحيه طريقه كتابه الأوامر و تنسيق الأكواد.



تم أطلاق أول لوحه تطويريه لأردوينو فى اواخر عام 2005 و أعتمدت على شريحه ATmega168 من العائله المشهوره AVR و التى تنتجها شركه Atmel للشرائح الألكترونيه ، و سميت هذه اللوحه بأسم Arduino Serial V.1

بعد أطلاق اللوحه قام مؤسسى المشروع بنشر كل ما يتعلق بأردوينو برخصه مفتوحه المصدر على موقع arduino.cc و الذى كان يعنى أن اى فرد فى العالم يمكنه الاطلاع و التعديل على التصميمات الهندسية و الشفرات المصدرية Source Codes لكل من بوردات اردوينو المختلفة Arduino Boards و البرمجيات Arduino IDE و تم نشر كل هذا مجاناً.

و ما ان تم نشر المشروع حتى توالت الكثير من التطويرات على التصميمات الهندسيه للوحات الألكترونيه و بيئه التطوير من مهندسين و هواه مختلفين حول العالم و تم أصدار العديد من اللوحات الأحدث مثل:

**Arduino UNO , Arduino Mega, Arduino lilyPad, Arduino Duemilanove, Arduino Fio**

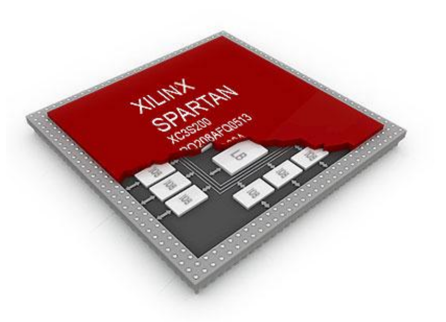
و كان القاسم المشترك فى كل الأصدارات السابقه هى انها تعتمد على شرائح المتحكمات الدقيقه من عائله AVR التى تنتجها شركه Atmel و حقق المشروع نجاح باهر حتى وصل عدد اللوحات الألكترونيه المباعه الى 300,000 قطعه فى مختلف دول العالم.




**فى عام** 2009 تعرض اردوينو لنقله نوعيه جديده جائت على يد فريق LeafLabs و الذى قام بتطوير لوحه أردوينو جديده تسمى The Maple board تعمل بشرائح ألكترونيه من عائله ARM Cortex و التى نقلت اردوينو الى افق جديد من السرعه و القوه حيث تتميز جميع شرائح ARM باحتوائها على معالجات عاليه السرعه تمتلك قوه معالجه تبدأ من 72 ميجا هرتز و تصل فى بعض الاصدارت الى 1.5 جيجا هرتز .

وبذلك أستطاع فريق LeafLabالخروج من حدود امكانيات شرائح AVR و استخدام الجيل الجديد من ARM Cortexمع الاحتفاظ بلغه أردوينو البرمجيه التى تتميز بالسهوله و خلوها من التعقيدات و فى ذات الوقت الاحتفاظ بعامل السعر حيث بلغ سعر Maple Board حوالى 34 دولار مع تكلفه الشحن.

**و بالرغم من كل** التطورات السريعه فى لوحات أردوينو المختلفه الا ان التطوير لم يتوقف عن حدود ARM Cortex بل وصل الى شرائح " مصفوفات البوابات المنطقية القابلة للبرمجة " Field-Programmable Gate Arrays (FPGA) فى عام 2010 تم نشر مشروع مفتوح المصدر اسمه Papilio Arduino و الهدف منه هو صناعه لوحه اردوينو تعمل بشرائح من نوع FPGA و يمكن برمجتها بلغه Arduino C




**تتميز شرائح** FPGA بالسرعه الفائقه فى اداء عمليات المعالجه بصوره متوازيه و فك تشفير البيانات Decoding لذلك تستخدم فى الكثير من التطبيقات الصناعيه التى يكون فيها عامل الوقت امرا حاسماً فى تصميم المنتج النهائى لكن يعيب هذه الشرائح عده أمور تجعلها مقتصره على المحترفين و الشركات فقط،

الأول: تكلفه لوحات التطوير الخاصه بها و التى يبدأ سعرها غالبا من 100 دولار و اكثر،

ثانيا: صعوبه برمجه الشرائح حيث تستخدم لغات برمجيه معقده فى تشغيل هذه الشرائح مثل VHDL و لغه Verilog .



جاء مشروع Papilio Arduino ليحل هذه المشكلات و يقدم لوحه تطوير جديده الى عائله لوحات اردوينو و تم اصدار Papilio Arduino IDE و هى نسخه معدله من برنامج Arduino IDE لجعل لغه اردوينو

تتناسب مع شرائح FPGA كما تم اصدار لوحات papilio boards للبيع من منتصف عام 2010 بأسعار تبدأ من 40 دولار فقط و يمكنك الاطلاع على المزيد من المعلومات عن لوحات papilio من الموقع التالى http://papilio.cc

بعد مشروع Papilio Arduino اصبحت عائله اردوينو تحتوى على لوحات تطويرات تعمل تقريبا بكافه التقنيات التى تم ابتكارها مثل AVR و شرائح ARM Cortex و حتى شرائح FPGA

و فى النهايه يمكننا القول أن مشروع اردوينو صنع أكبر مجتمع تقنى من المهندسين و الفنين و الهواه يعملون على تطوير الافكار و المشاريع المتعلقه بالتحكم الآلى بصوره تشاركيه و تفاعليه حول العالم اعتماداً على استخدام لوحات الكترونيه بتقنيات مخلتفه و مع ذلك يتم برمجتها جميعا بلغه برمجيه واحده و مجانيه و متاحه للجميع.

انه مجتمع جديد و مختلف .. حقاً انها ثوره الكترونيه ....




## فاب لاب مصر او ما يعرف بأسم ورشه التصنيع الشخصي

هو مكان مُصمم لتحويل الافكار من مجرد خيال الى واقع ملموس و بناء مجتمع من مصنعى الافكار و المنتجات من الهواه و المتحرفين فى مجالات مختلفه مثل: (الالكترونيات – الهندسه الميكانيكيه – هندسه الحاسب – التصنيع الرقمى – الفنون)

و يوفر المكان العديد من الادوات التقنيه التى تساعدك على تصنيع افكارك مثل:

- ماكينه القطع بالليزر LASER cutter machine
- طابعه ثلاثيه الابعاد 3D printer
- ماكينه حفر بالتحكم الرقمى CNC machine
- بوردات اردوينو المختلفه و مكونات الكترونيه منوعه

و العديد من الادوات الاخرى ....

يهدف الفاب لاب الى توفير بيئه تشاركيه و تعليميه فى ذات الوقت تساعد الافراد على الابداع و اطلاق العنان لافكارهم و خيالهم و مشاركه هذه الافكار مع المجتمع.

كما يوفر وورش و دورات تدريبيه للافراد مثل:

- ورش تدريب عمليه لاستخدام اردوينو Arduino workshops
- ورش تدريبيه للاطفال لتعلم الالكترونيات Young Fab.Academy

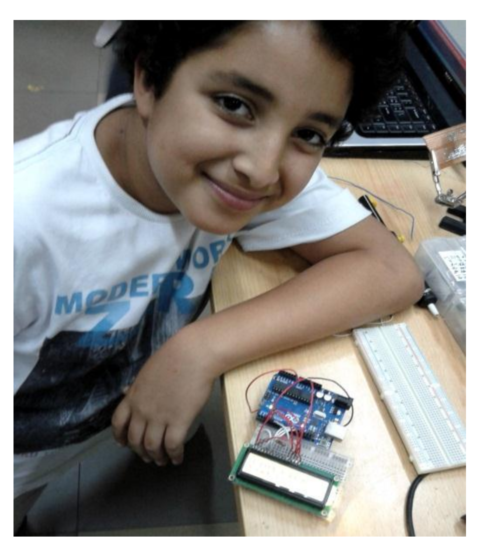
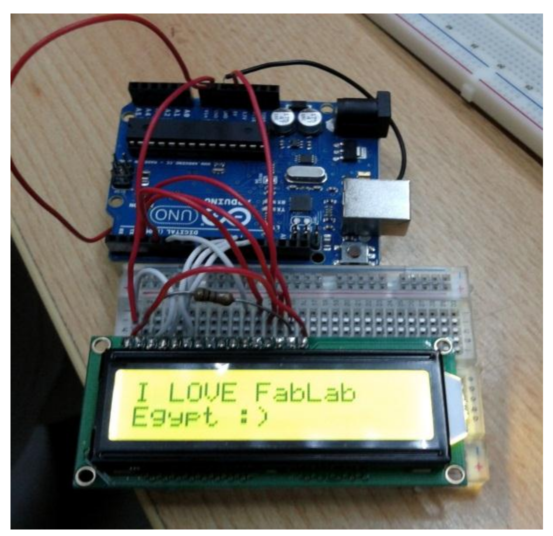
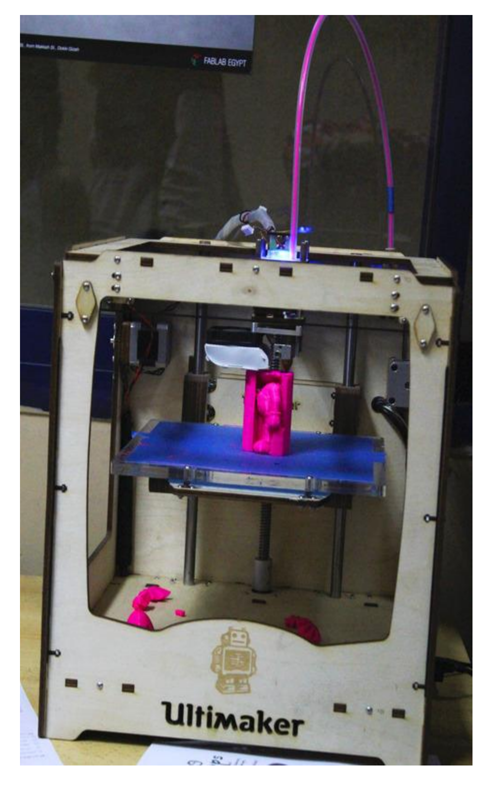


اذا كانت لديك فكره و تريد ان تنفذها و تحتاج الى بعض المساعده او تعاون من اشخاص اخريين او تحتاج الى دورات تدريبيه فى اردوينو فيمكنك زياره فاب لاب مصر يوميا من الساعه 12 ظهرا حتى 8 مسائاً جميع الايام

http://fablab-egypt.com

https://www.facebook.com/fablab.egypt


52



I LOVE FabLab
Egypt :)

FABLAB EGYPT
Ultimaker






## المصادر التعليميه التى اعتمدت عليها فى كتابه محتوى الكتاب:


- http://www.instructables.com/id/Arduino-Projects
- http://www.oomlout.com
- http://ladyada.net
- http://bildr.org
- http://hlt.media.mit.edu/?cat=5
- http://circuit-projects.com/microcontroller
- http://www.eeweb.com
- http://www.lvl1.org
- http://www.coolcircuit.com
- http://www.instructables.com/id/Breadboard-How-To
- http://blog.makezine.com/2009/12/11/arduino-shields-open-source-hardwar/

## لمن يريد أستخدام اردوينو مع الماتلاب MATLAB

- http://www.mathworks.com/academia/arduino-software/arduino-matlab.html
- http://arduino.cc/playground/Interfacing/Matlab

## مشاريع رائعه بأستخدام اردوينو يمكنك ان تصنعها بنفسك:

- http://fritzing.org/projects/
- http://www.instructables.com/id/Arduino-Projects
- http://www.ladyada.net/make/boarduino/index.html
- http://www.ladyada.net/make/mshield/index.html
- http://www.shapeoko.com/wiki/index.php/About
- http://www.ladyada.net/make/monochron/index.html
- http://www.ladyada.net/make/bedazzler/index.html
- http://www.ladyada.net/make/mintyboost/index.html
- http://dangerousprototypes.com/forum/viewtopic.php?f=56&t=2892#p28410
- http://www.ladyada.net/make/logshield/
- http://www.wayneandlayne.com/projects/video-game-shield/
- http://excamera.com/sphinx/gameduino/
- http://dangerousprototypes.com/2012/03/24/arduino-voice-control-with-easyvr-shield/
- http://blog.minibloq.org/p/documentation.html
- http://www.doctormonk.com/2011/09/arduino-solar-radio.html
- http://j4mie.org/blog/how-to-make-a-physical-gmail-notifier/
- http://blog.tinyenormous.com/2008/11/25/gmail-notifier-project-for-dummies/
- http://reprap.org/wiki/Arduino_Mega_Pololu_Shield

# مراجع أخرى: كُتب


- 30 Arduino Evil Genuis projects
- Arduino.Internals
- Arduino Projects to Save the World
- Arduino robotics
- Beginning Android ADK with Arduino
- Environmental Monitoring Arduino
- Getting Started with Arduino (Second Edition)
- Shrinkify Your Arduino Projects -burn arduino to ATiny (Video)
- Make a mind-controlled arduino robot
- Make_Magazine_Volume_25
- Making Things Talk Using Sensors (Second Edition)
- Making Things see hear and feel your world (Second Edition)
- O'Reilly - Arduino Cookbook
- O'Reilly-Make - Arduino Bots and Gadgets
- Oreilly Verlag - Arduino Physical Computing
- Practical Arduino Cool Projects For Open Source Hardware
- Programming Interactivity
- Programming Your Home Automate with Arduino, Android, and Your Computer
- The Art of Electronics 2nd edition Complete - Horowitz and Hill